

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP HCM
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



KHÓA LUẬN TỐT NGHIỆP

THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION
CHỈNH SỬA VIDEO

Giáo viên hướng dẫn: TS. Huỳnh Xuân Phụng

Sinh viên thực hiện: Trần Quốc Việt

MSSV: 18110238

Khóa: K18

Ngành: Công Nghệ Thông Tin

TP. Hồ Chí Minh, ... tháng 6 năm 2023

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP HCM
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



KHÓA LUẬN TỐT NGHIỆP

THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION
CHỈNH SỬA VIDEO

Giáo viên hướng dẫn: TS. Huỳnh Xuân Phụng

Sinh viên thực hiện: Trần Quốc Việt

MSSV: 18110238

Khóa: K18

Ngành: Công Nghệ Thông Tin

TP. Hồ Chí Minh, ... tháng 6 năm 2023



TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
**KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO**
www.fhq.hcmute.edu.vn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---***---

TP. Hồ Chí Minh, ngày ... tháng 6 năm 2023

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ và tên Sinh viên: Trần Quốc Việt

MSSV: 18110238

Ngành: Công Nghệ Thông Tin

Lớp: 18110ST1

Giảng viên hướng dẫn: TS. Huỳnh Xuân Phụng

Ngày nhận đề tài:..... **Ngày nộp đề tài:**.....

1. **Đề tài:** THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO
2. **Nội dung thực hiện đề tài:**
 - Tìm hiểu các tác vụ cơ bản của việc chỉnh sửa video.
 - Tìm hiểu về giới hạn của web application khi thực hiện việc chỉnh sửa video
 - Tìm hiểu và thiết kế giao diện một web application bằng framework Angular
 - Tìm hiểu về các công cụ cho phép thao tác lên video
 - Tìm hiểu giới hạn và xây dựng hệ thống có thể rendering video bằng NodeJS.
3. **Sản phẩm:** -WebApplication có thể thực hiện được các thao tác chỉnh sửa video thông dụng và hệ thống rendering video.

TRƯỞNG NGÀNH

GIÁO VIÊN HƯỚNG DẪN



TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM

**KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO**

www.fhq.hcmute.edu.vn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---***---

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên Sinh viên: Trần Quốc ViệtMSSV: 18110238

Tên đề tài: THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO

Họ và tên Giáo viên hướng dẫn: TS. Huỳnh Xuân Phụng.....

NHẬN XÉT

1. Về nội dung đề tài và khối lượng thực hiện:

.....

2. Ưu điểm:

.....

3. Nhược điểm:

.....

4. Đề nghị cho bảo vệ hay không?

.....

5.Diễn

.....(Bằng chữ:)

TP. Hồ Chí Minh, ngày ... tháng 6 năm 2023

Giáo viên hướng dẫn

(Ký tên & ghi rõ họ tên)



TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO
www.fhq.hcmute.edu.vn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---***---

NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên Sinh viên: Trần Quốc Việt MSSV: 18110238

Tên đề tài: THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO

Họ và tên Giáo viên phản biện: Ths. Mai Anh Thơ.....
.....

NHẬN XÉT

1. Về nội dung đề tài và khối lượng thực hiện:

.....
.....

2. Ưu điểm:

.....
.....

3. Nhược điểm:

.....
.....

4. Đề nghị cho bảo vệ hay không?

.....

5.Diễn

.....(Bằng chữ:)

TP. Hồ Chí Minh, ngày ... tháng 6 năm 2023

Giáo viên phản biện

(Ký tên & ghi rõ họ tên)

LỜI CẢM ƠN

Đầu tiên em xin gửi lời cảm ơn đến thầy cô và lãnh đạo của Trường Đại học Sư phạm và Kỹ Thuật TP. Hồ Chí Minh, đã tạo điều kiện thuận lợi về học thuật và cơ sở vật chất cho em có thể học tập, rèn luyện, củng cố kiến thức và phát triển bản thân trong suốt thời gian em theo học tại trường.

Nhờ sự giúp đỡ từ các giáo viên của khoa Công nghệ thông tin cũng như khoa Đào tạo Chất Lượng Cao nói chung, và giáo viên hướng dẫn của em, thầy Huỳnh Xuân Phụng, nói riêng đã luôn theo sát, luôn giải đáp thắc mắc và hỗ trợ em đưa ra các hướng giải quyết các vấn đề khó khăn em gặp phải trong khoảng thời gian em thực hiện đề tài. Em thật sự chân thành gửi lời cảm ơn đến với các thầy cô.

Cùng với đó em xin gửi lời cảm ơn đến bạn bè đã cùng em tham gia học tập, đặc biệt là các bạn ở lớp CL2B, đã cùng em cố gắng học tập để có thể đi tới thời điểm hiện tại. Ngoài ra, em cũng muốn gửi lời cảm ơn đến các anh chị Ban Truyền thông Khoa Chất Lượng Cao, đã tạo điều kiện cho em tiếp xúc với nhiều vấn đề ngoài học thuật chuyên môn, từ đó cho em nhiều góc nhìn về phương pháp giải quyết vấn đề.

Và cuối cùng là lời cảm ơn gửi tới gia đình đã luôn tin tưởng, đặt niềm tin vào con đường học thuật mà em chọn và luôn động viên em trong khoảng thời gian làm đề tài này.

Em xin chân thành cảm ơn.

TÓM TẮT

Hiện tại dưới sự phát triển nhanh chóng của mạng Internet, đặc biệt là sau lần bùng phát dịch COVID2019 vào hai quý cuối năm 2021, số lượt truy cập vào các mạng xã hội từ nổi tiếng cho tới những cái tên xa lạ tăng nhanh, ta có thể dễ dàng thấy được điều này thông qua việc các trang tin, báo đài trong thời gian này đề cập tới các mạng xã hội thuần Việt như Lotus, Butta, Gapo, ... Tuy có sự tăng trưởng lại tương đối, nhưng trong thời điểm này Tiktok là một trong những cái tên bùng nổ, với danh hiệu mạng xã hội chia sẻ video lớn nhất thế giới vượt qua cả Youtube. Như Isaac Newton từng nói “Nếu tôi nhìn xa hơn đó là vì tôi đang đứng trên vai người khổng lồ”, vì vậy thay vì tạo ra các mạng xã hội và chen chúc với các ông lớn trong ngành thì em hướng tới các công cụ hỗ trợ cho họ và thể ý tưởng về một “Web Application chỉnh sửa video” được ra đời. Web Application này sẽ hướng tới các đối tượng người dùng mới bắt đầu, với cách thức làm việc tối giản để đạt hiệu quả về hiệu năng tốt nhất

MỤC LỤC

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	I
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	II
NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN	III
LỜI CẢM ƠN.....	IV
TÓM TẮT.....	V
MỤC LỤC	VI
DANH MỤC BẢNG	X
DANH MỤC HÌNH	XII
DANH MỤC TỪ VIẾT TẮT	XIII
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI.....	1
1.1. LÝ DO CHỌN ĐỀ TÀI.	1
1.2. MỤC TIÊU.	1
1.3. CÔNG NGHỆ SỬ DỤNG.	1
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	2
2.1. NODEJS (BACK-END).	2
2.1.1. Giới thiệu.....	2
2.1.2. Ưu điểm.....	2
2.1.3. Nhược điểm.....	2
2.1.4. Lý do sử dụng.....	2
2.1.5. Các thư viện được sử dụng.....	3
2.2. ANGULAR (FRONT-END).....	5
2.2.1. Giới thiệu.....	5
2.2.2. Ưu điểm.....	5
2.2.3. Nhược điểm.....	5
2.2.4. Lý do sử dụng.....	5
2.2.5. Các thư viện được sử dụng.....	5
2.3. FFMPEG.....	6
2.3.1. Giới thiệu.....	6
2.3.2. Ưu điểm.....	6

2.3.3.	Nhược điểm.....	6
2.3.4.	Lý do sử dụng.....	6
CHƯƠNG 3. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU.....		7
3.1.	KHẢO SÁT HIỆN TRẠNG.....	7
3.1.1.	Trang web Clipchamp.com.....	7
3.1.2.	Phần mềm DaVinci Resolve.....	8
3.2.	XÁC ĐỊNH YÊU CẦU.....	9
3.2.1.	Nghiệp vụ.....	9
3.2.1.1.	Đối tượng sử dụng.....	9
3.2.1.2.	Chức năng.....	9
3.3.	ĐÁNH GIÁ VÀ LỰA CHỌN CÔNG NGHỆ.....	9
3.3.1.	Mục tiêu đề tài.....	9
3.3.2.	Nội dung.....	9
3.3.3.	Môi trường lập trình.....	10
3.3.4.	Phương pháp nghiên cứu.....	10
3.3.5.	Ý nghĩa khoa học, thực tiễn.....	10
CHƯƠNG 4. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....		11
4.1.	PHÂN TÍCH HỆ THỐNG.....	11
4.1.1.	Sơ đồ lớp.....	11
4.1.2.	Sơ đồ usecase.....	12
4.1.3.	Mô tả sơ đồ usecase.....	15
4.1.4.	Đặc tả Usecase.....	18
4.1.4.1.	Usecase đăng ký.....	18
4.1.4.2.	Usecase đăng nhập.....	18
4.1.4.3.	Usecase đăng tải video.....	19
4.1.4.4.	Usecase tạo điểm cắt video.....	20
4.1.4.5.	Usecase gửi yêu cầu render.....	22
4.1.4.6.	Usecase nhận video từ client.....	22
4.1.4.7.	Usecase ghép video.....	23
4.1.4.8.	Usecase render video.....	24

4.1.4.9.	Usecase addLUTFilter.....	26
4.1.4.10.	Usecase send video to e-mail.....	26
4.1.4.11.	Usecase xóa dữ liệu cũ.....	27
4.2.	THIẾT KẾ HỆ THỐNG	28
4.2.1.	Tổng quát chương trình.....	30
4.2.2.	Tổng quan về Database	31
4.3.	THIẾT KẾ GIAO DIỆN	33
4.3.1.	Danh sách giao diện người dùng.....	33
4.3.1.1.	Mô tả chi tiết giao diện.....	33
4.3.2.	Danh sách giao diện chương trình chính.....	38
4.3.2.1.	Giao diện chung.....	38
4.3.2.2.	Mô tả chi tiết giao diện.....	40
4.4.	THIẾT KẾ XỬ LÝ DỮ LIỆU ĐOẠN VIDEO	49
4.4.1.	Thiết kế cách thức cắt video.....	49
4.4.2.	Thiết kế cách thức ghép video	50
4.4.3.	Thiết kế phương thức tăng tốc video	51
4.4.4.	Thiết kế phương thức thay đổi màu sắc video	52
CHƯƠNG 5.	CÀI ĐẶT VÀ KIỂM THỬ	53
5.1.	CÀI ĐẶT.....	53
5.1.1.	Cài đặt môi trường	53
5.1.2.	Cài đặt phần mềm.....	53
5.2.	KIỂM THỬ.....	54
5.2.1.	Kiểm thử chức năng login.....	54
5.2.2.	Kiểm thử chức năng sign up	55
5.2.3.	Kiểm thử chức năng upload	55
5.2.4.	Kiểm thử chức năng send request và render video.	57
CHƯƠNG 6.	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	58
6.1.	TỔNG KẾT.....	58
6.2.	KẾT LUẬN.....	58
6.2.1.	Về mặt lý thuyết.....	58

6.2.2.	Về mặt ứng dụng	58
6.2.3.	Ưu điểm.....	58
6.2.4.	Nhược điểm.....	58
6.3.	HƯỚNG PHÁT TRIỂN.....	59
TÀI LIỆU THAM KHẢO		60

DANH MỤC BẢNG

Bảng 1 Bảng thông tin về các thư viện được sử dụng tại back-end.	3
Bảng 2 Bảng thông tin các thư viện được sử dụng tại front-end	5
Bảng 3: Bảng mô tả nội chi tiết dung sơ đồ usecase	15
Bảng 4 Bảng mô tả chi tiết usecase đăng ký	18
Bảng 5 Bảng mô tả chi tiết usecase đăng nhập	18
Bảng 6 Bảng mô tả chi tiết usecase đăng tải video	19
Bảng 7 Bảng mô tả chi tiết usecase tạo điểm cắt video	20
Bảng 8 Bảng mô tả chi tiết usecase yêu cầu render.	22
Bảng 9 Bảng mô tả chi tiết usecase nhận video từ client	22
Bảng 10 Bảng mô tả chi tiết usecase ghép video	23
Bảng 11 Bảng mô tả chi tiết usecase render video.....	24
Bảng 12 Bảng mô tả chi tiết usecase addLUTFilter.....	26
Bảng 13 Bảng mô tả chi tiết usecase send video to email.....	26
Bảng 14 Bảng mô tả chi tiết usecase xóa dữ liệu cũ	27
Bảng 15 Bảng thể hiện chi tiết bảng Users.....	31
Bảng 16 Bảng thể hiện chi tiết bảng Project_data.....	32
Bảng 17 Bảng chi tiết giao diện trang chủ	33
Bảng 18 Bảng chi tiết giao diện login	34
Bảng 19 Bảng chi tiết giao diện đăng ký.....	35
Bảng 20 Bảng chi tiết giao diện dashboard	36
Bảng 21 Bảng mô tả giao diện chung.....	38
Bảng 22 Bảng mô tả chi tiết giao diện đăng tải nội dung video.....	40
Bảng 23 Bảng mô tả chi tiết giao diện video	42
Bảng 24 Bảng mô tả chi tiết giao diện danh sách các điểm cắt	44
Bảng 25 Bảng mô tả chi tiết giao diện ghép nội dung video.....	45
Bảng 26 Bảng mô tả chi tiết giao diện modal setting.....	47
Bảng 27 Bảng mô tả chi tiết thiết kế cách thức cắt video	49
Bảng 28 Bảng mô tả chi tiết thiết kế cách thức ghép video	50
Bảng 29 Bảng mô tả chi tiết thiết kế cách thức tăng tốc video	51

Bảng 30	Bảng mô tả chi tiết thiết kế cách thức thay đổi màu sắc video.....	52
Bảng 31	Bảng kiểm thử chức năng login	54
Bảng 32	Bảng kiểm thử chức năng sign up.....	55
Bảng 33	Bảng mô tả kiểm thử chức năng upload video.....	55
Bảng 34	Bảng kiểm thử chức năng send request và render video.....	57

DANH MỤC HÌNH

Hình 1 Hình ảnh mô hình áp dụng.	2
Hình 2 Hình ảnh chương trình clipchamp tại trang clipchamp.com	7
Hình 3 Hình ảnh phần mềm DaVinci Resolve 18	8
Hình 4 Hình ảnh về sơ đồ lớp.....	11
Hình 5 Hình ảnh Usecase của khách	12
Hình 6 Hình ảnh Usecase của người dùng	13
Hình 7 Hình ảnh usecase của máy.....	14
Hình 8 Hình ảnh lược đồ tuần tự chức năng đăng tải video.....	19
Hình 9 Hình ảnh lược đồ tuần tự tạo thông tin cắt ghép	21
Hình 10 Hình ảnh lược đồ tuần tự xử lý video.....	25
Hình 11 Hình ảnh lược đồ hoạt động của chức năng đăng tải video	28
Hình 12 Hình ảnh lược đồ hoạt động của chức năng xử lý video	29
Hình 13 Hình ảnh thể hiện cách chuyển trang trong chương trình	30
Hình 14 Hình ảnh sơ đồ ER của database	31
Hình 15 Hình ảnh giao diện trang chủ.....	33
Hình 16 Hình ảnh giao diện login	34
Hình 17 Hình ảnh giao diện đăng ký.....	35
Hình 18 Hình ảnh giao diện dashboard	36
Hình 19 Hình ảnh giao diện Page Not Found	37
Hình 20 Hình ảnh giao diện chung của chương trình.....	38
Hình 21 Hình ảnh giao diện chung của chương trình.....	38
Hình 22 Hình ảnh giao diện chi tiết của giao diện đăng tải nội dung video	40
Hình 23 Hình ảnh giao diện chi tiết của giao diện video	41
Hình 24 Hình ảnh giao diện chi tiết của giao diện video	41
Hình 25 Hình ảnh giao diện chi tiết của giao diện danh sách điểm cắt	44
Hình 26 Hình ảnh giao diện chi tiết của giao diện ghép nội dung video	45
Hình 27 Hình ảnh giao diện chi tiết của giao diện modal setting	47

DANH MỤC TỪ VIẾT TẮT

UI: User-Interface

CPU: Central Processing Unit

UX: User-Experience

Web: Website

CORS: Cross-origin resource sharing

Webapp: Web Application

HTML: Hyper-Text Markup Language

CSS: Cascading Style Sheet

JS: JavaScript

BE: Back-End

UC: Use-case

OTP: One time Password

SFolder: Session Folder

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1. LÝ DO CHỌN ĐỀ TÀI.

Trong những năm gần đây, dưới sự phát triển của công nghệ, đặc biệt là mạng xã hội, con người truy cập internet nhằm mục đích kết nối với nhau trở nên nhiều hơn. Có thể nói là nhu cầu này không thể giảm mà chỉ có thể tiếp tục tăng lên, ta có thể dễ dàng nhận ra được việc này thông qua việc các trang tin đưa tin thông qua mạng xã hội này càng nhiều hơn. Trong đó các bài đăng bằng video ngắn ngày càng phát triển, có thể lấy Tiktok ra làm một ví dụ điển hình cho sự phát triển này, theo tin tức của tờ ICTNews cho thấy tỉ lệ người dùng sử dụng ứng dụng này tăng từ 34% (2020) lên 53% (2021); thời lượng sử dụng ứng dụng thì tăng gấp đôi từ 4% lên 8% (2021). Ngoài ra thì có thể thấy được hai ông lớn trong làng công nghệ là Meta và Alphabet (Google) đã cập nhật tính năng chia sẻ video ngắn cho hai nền tảng xã hội của họ là Facebook Reel và Youtube Short.

Nắm bắt được xu hướng thị trường, thay vì tạo ra một nền tảng mạng xã hội để cạnh tranh với các ông lớn, em đã nghiên cứu phát triển nên một phần mềm nói đúng hơn là một web application nhằm mục đích tạo ra các video ngắn để người dùng cơ bản có thể sử dụng và tạo ra các video của riêng họ và đăng tải nó lên các mạng xã hội

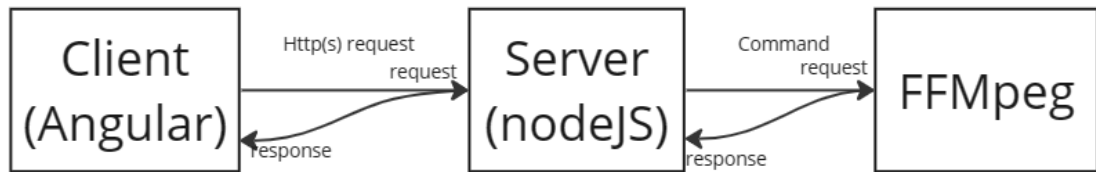
1.2. MỤC TIÊU.

- Xây dựng trang web cho phép người dùng chỉnh sửa các video của họ upload lên, như cắt ghép và hiệu chỉnh một vài thông số cơ bản của một video, hướng tới nhóm người dùng mới bắt đầu vào cũng đang bắt đầu quen dần với internet
- Giao diện trang web hướng tới sự đơn giản, dễ thao tác và dễ sử dụng.
- Nghiên cứu về thao tác người dùng được áp dụng để có thể tận dụng tối đa sự quen biết của người dùng

1.3. CÔNG NGHỆ SỬ DỤNG.

- Back-end (server): chương trình sử dụng NodeJS cụ thể hơn là ExpressJS để tạo ra các point để thực hiện các yêu cầu từ client.
- Front-end (client): sử dụng Angular để tạo nên giao diện người dùng.
- Xử lý video: ffmpeg (ganya.dev).

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT



Hình 1 Hình ảnh mô hình áp dụng.

2.1. NODEJS (BACK-END).

2.1.1. Giới thiệu.

NodeJS được xây dựng dựa trên JavaScript runtime, nhưng về phần lõi thì NodeJS được viết dựa trên C++ và Javascript. Được sử dụng tại phần BE nhằm tạo ra môi trường để làm việc trực tiếp với máy tính là trở thành công cụ giao tiếp với phía Client.

2.1.2. Ưu điểm.

Về ưu điểm như tên gọi của nó JS, nó cho phép người dùng sử dụng JavaScript là ngôn ngữ lập trình chính. Cùng với đó NodeJS cho phép thực hiện các sự kiện bất đồng bộ, từ đó có thể xử lý nhiều việc cùng một lúc. Ngoài ra thì NodeJS sở hữu một lượng lớn các package giúp lập trình viên dễ dàng sử dụng.

2.1.3. Nhược điểm.

Có thể nói javascript cũng là một trong những rào cản, vì tuy đơn giản nhưng javascript sẽ trở nên rất phức tạp và khó chịu nếu thực hiện các công việc liên quan tới nhiều CPU. Ngoài ra trong khi sử dụng các vấn đề rắc rối như callback hell.

2.1.4. Lý do sử dụng.

Về lý do sử dụng, NodeJS có một cộng đồng người dùng thật sự lớn và cộng đồng sửa lỗi cũng được coi là khá lớn, ngoài ra dùng NodeJS có thể giúp bản thân chương trình được đồng bộ về mặt ngôn ngữ lập trình từ đầu tới cuối, từ đó có thể dễ dàng sửa chữa.

2.1.5. Các thư viện được sử dụng.**Bảng 1 Bảng thông tin về các thư viện được sử dụng tại back-end.**

STT	Tên thư viện	Lý do sử dụng
1	Express	Ưu điểm: đây là framework hướng tới sự tối giản nhẹ và nhanh. Tạo các Endpoint bằng framework này dễ dàng, cùng với đó ta có thể đặt các middleware vào API cũng đơn giản. Ngoài ra nhóm phát triển cũng thuộc OpenJS Foundation nên về bản chất ExpressJS sinh ra là cho NodeJS
2	Nodemon	Đây là một thư viện hỗ trợ cho việc phát triển phần mềm, mỗi lần ta save lại nó sẽ tự động restart lại project từ đó có thể rút ngắn được thời gian lập trình
3	Body parser	Là một middleware làm tối giản lại các dữ liệu được gửi thông qua body trên request được gửi từ client
4	Cors	Là middleware giúp các tài nguyên được chia sẻ chéo. Làm client và server có thể giao tiếp với nhau và đảm bảo được lỗi bảo mật CORS
5	Dotenv	Là thư viện để tạo ra các biến ảo để thiết lập môi trường
6	Multer	Là thư viện giúp chuyển dẫn các file giữa client và server
7	Short unique id	Là một thư viện tạo ra các định danh độc nhất ngắn, giúp phân biệt giữ các file được sinh ra trong một session
8	Nodemailer	Là thư viện giúp người dùng gửi mail một cách tự động, ngoài ra thư viện này hỗ trợ rất nhiều các dịch vụ mail thông dụng để dễ dàng thực hiện gửi mail
9	Cloudinary	Là một thư viện hỗ trợ môi trường giao tiếp với dữ liệu được lưu trên hệ thống của Cloudinary
10	Json web token	Là thư viện tạo ra các token, những token này dùng trong việc xác thực danh tính và giới hạn thời gian đăng nhập

11	Bcrypt	Là thư viện được dùng trong việc băm mật khẩu khi lưu vào database, mặc dù các mật khẩu chỉ dùng một lần nhưng việc này sẽ đảm bảo được bảo mật cơ bản
12	Mysql	Là thư viện giúp nodeJS có thể dễ dàng kết nối tới mysql database dễ dàng
13	Node-cmd	Là thư viện hỗ trợ việc dùng các câu lệnh cmd để tương tác với máy tính.

2.2. ANGULAR (FRONT-END).

2.2.1. Giới thiệu.

Angular là một nền tảng và front-end framework để có thể tạo nên các single-page client application sử dụng HTML và Typescript.

2.2.2. Ưu điểm.

Về ưu điểm, xét về mặt lập trình, angular sử dụng typescript, đây là một ngôn ngữ được dựa trên javascript, nó khắc phục được phần lớn các điểm yếu của javascript. Ngoài ra cấu trúc của một project angular là một ưu điểm, với việc chia các file ra riêng biệt, từ đó việc viết các đoạn code HTML hoặc CSS sẽ trở nên quen thuộc hơn.

2.2.3. Nhược điểm.

Về nhược điểm, tuy TypeScript là một ngôn ngữ rất tốt, nhưng vì nó là một phiên bản của JavaScript nên việc cập nhật các function mới của JS lên TS sẽ nhiều lúc bị chậm trễ.

2.2.4. Lý do sử dụng.

Angular được chọn trong project lần này vì đây là một project khá là đặt thù. Có thể nói chương trình này là một phần mềm chỉnh sửa video được tối ưu để sử dụng trên web, vì vậy ta cần một framework front-end hoàn chỉnh và Angular có thể giải quyết được vấn đề này. Ngoài ra em có kinh nghiệm sử dụng Angular, nên sẽ tốt khi có nhiều vấn đề cần phải giải quyết.

2.2.5. Các thư viện được sử dụng.

Bảng 2 Bảng thông tin các thư viện được sử dụng tại front-end

STT	Tên thư viện	Lý do sử dụng
1	Angular 14 Framework pack	Đây là framework frontend hoàn chỉnh, có mọi thứ người dùng cần, trong đó, từ việc định hướng cho tới tích hợp các gói animation hỗ trợ giao diện
2	File saver	Là một thư viện hỗ trợ việc download các file được gửi từ server
3	Angular2 uuid	Là thư viện giúp tạo ra các id được dùng để hỗ trợ việc xác định các object trong mảng

2.3. FFMPEG.

2.3.1. Giới thiệu.

FFmpeg là một phần mềm open source cho các tác vụ multimedia. Chương trình này có các thư viện và các phần mềm để có thể thao tác lên video, audio và những tệp đa phương tiện và phát sóng trực tiếp. Bộ công cụ này hiện có thể chạy được trên các hệ điều hành lớn như Windows, MacOS và Linux. Được ra đời vào cuối năm 2000, với mục đích chỉ để cắt, ghép video, hiện tại phần mềm này đã có tuổi đời 21 năm, phát triển thêm nhiều tính năng như scaling, compress, filtering, ..., trong thời gian phát triển đó có rất nhiều phần mềm nổi tiếng đã sử dụng bộ công cụ này, ta có thể kể tới như VLC (một phần mềm chơi đa phương tiện), hay được sử dụng trong lõi của việc xử lý video cho Youtube và Bilibili.

2.3.2. Ưu điểm.

Đầu tiên có thể nhắc đến đây là bộ công cụ mã nguồn mở, đồng nghĩa với việc nó miễn phí và có thể có rất nhiều biến thể từ những lập trình viên khác nhau. Ngoài ra kỹ thuật xử lý video nén của nó có tốc độ nhanh, và theo một số nhận xét thì, tốc độ encode với decode của bộ dụng cụ này nhanh hơn so với các phần mềm chỉnh sửa chuyên nghiệp của Adobe.

2.3.3. Nhược điểm.

Bộ công cụ này hiện tại không có một UI, nên để tiếp cận và học tập nó tất cả phải thông qua các câu lệnh thực hiện trên command prompt. Về tư liệu học tập cũng không phổ biến, ngoài các tài liệu được cung cấp bởi chính chủ ra thì gần như rất hiếm các tài liệu viết lại hoặc các bài giảng để việc học tập trở nên nhanh chóng. Việc này có lẽ nằm ở bản chất của chương trình, vì được tạo ra như một nền tảng xử lý nên người dùng trực tiếp trở nên quá ít, dẫn tới cộng đồng trở nên nhỏ và các câu hỏi đặc ra được ít trả lời.

2.3.4. Lý do sử dụng.

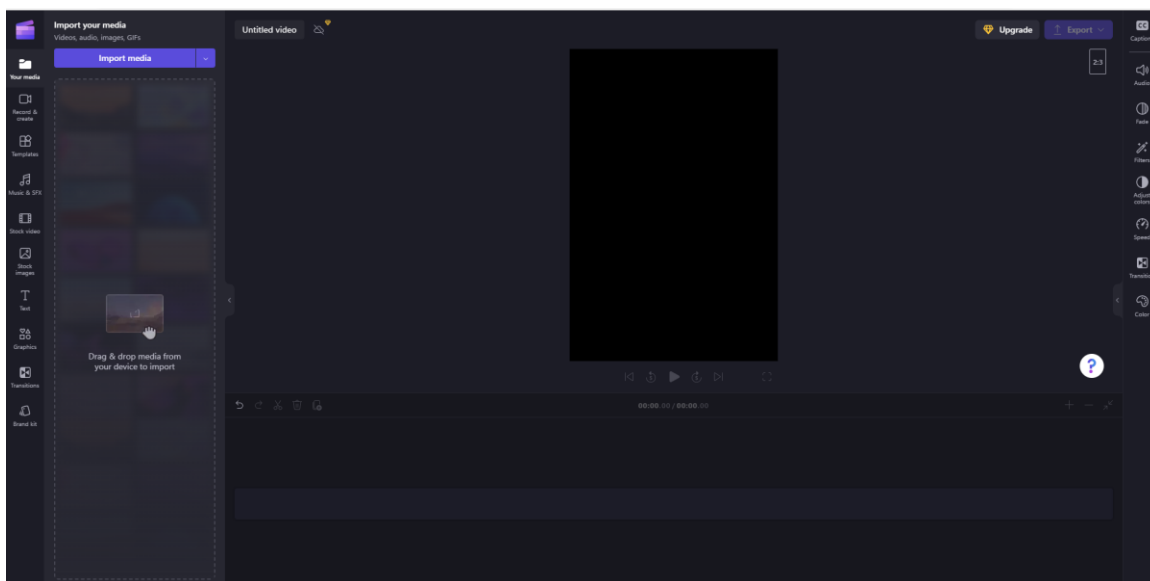
Như đã nhắc tới ở trên thì đây là bộ công cụ open source, từ đó có thể dễ dàng sử dụng và phát triển nó không cần quan tâm tới các vấn đề liên quan tới bản quyền chương trình. Ngoài ra như đã nhắc ở trên tốc độ xử lý là một trong những yếu tố được cân nhắc, vì phải xử lý thông qua server, từ đó giảm thời gian chờ của người dùng cũng là một điểm cần được chú ý.

CHƯƠNG 3. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

3.1. KHẢO SÁT HIỆN TRẠNG

3.1.1. Trang web Clipchamp.com

Trang web Clipchamp là một trang web cho phép người dùng đưa video của bản thân lên trang web này và thực hiện chỉnh sửa video gồm các thao tác cơ bản. Trang web này được phát triển triển và phát hành bởi đội ngũ của Microsoft, ngoài ra phiên bản desktop của trang web này được Microsoft phát hành kèm chung với Windows 11 để trở thành một trong những công cụ có sẵn của phiên bản windows này. Có thể thấy vì là được mới phát hành gần đây nhưng Clipchamp đã có một lượng người dùng tương đối và để lại đánh giá khá tốt tại thang điểm của google search.



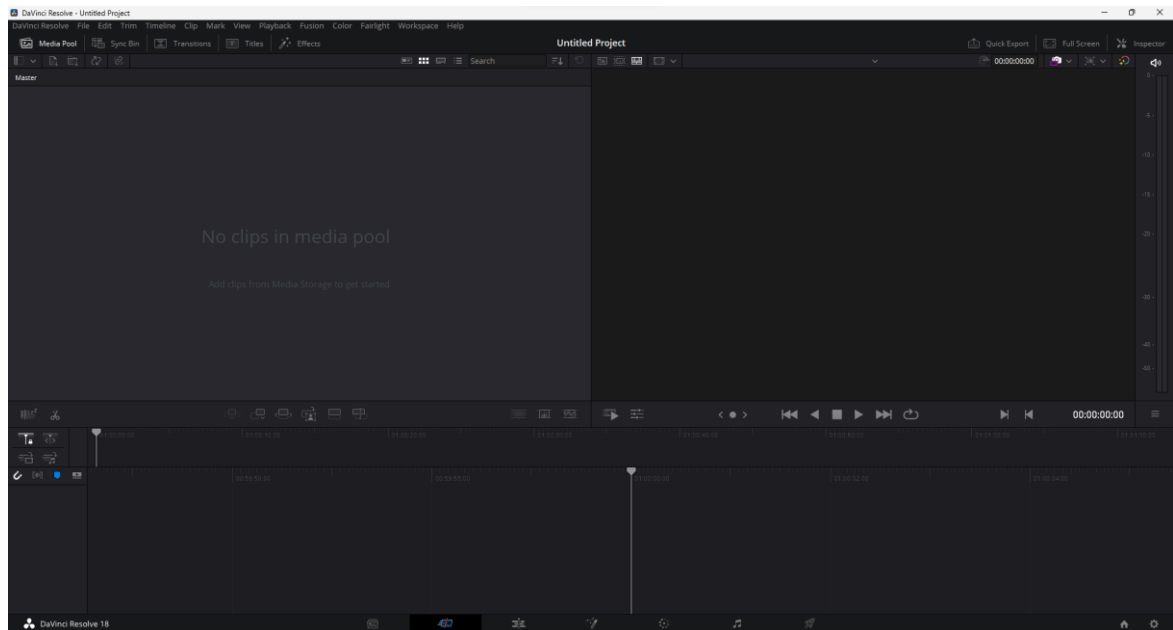
Hình 2 Hình ảnh chương trình clipchamp tại trang clipchamp.com

Ưu điểm

- Chương trình được thiết kế với phiên bản web dễ dàng tiếp cận.
- Giao diện hiện đại, có sự giao thoa giữa ngôn ngữ thiết kế của Windows 11 và Material Design.
- Các thao tác quen thuộc, khá giống với các chương trình chỉnh sửa video trên desktop.
- Có nhiều template được dựng sẵn để người dùng sử dụng.
- Phiên bản miễn phí khá đầy đủ tính năng

3.1.2. Phần mềm DaVinci Resolve

Phần mềm DaVinci Resolve là chương trình chỉnh sửa video chuyên nghiệp được BlackmagicDesign, một thương hiệu sản xuất ra các máy quay chuyên nghiệp, thực hiện và phát hành. Có thể nói đây là một trong những phần mềm cực kỳ nổi tiến trong cộng đồng editor, vì phiên bản miễn phí được công ti này trực tiếp phát hành và phiên bản miễn phí này cung cấp một lượng lớn công cụ mạnh mẽ để thực hiện các thao tác chỉnh sửa.



Hình 3 Hình ảnh phần mềm DaVinci Resolve 18

Ưu điểm:

- Phần mềm được phát hành miễn phí
- Bộ công cụ đa dạng và mạnh mẽ để thực hiện các thao tác chuyên nghiệp.
- Hệ thống cân chỉnh màu sắc cực kỳ mạnh mẽ, gồm nhiều thông số chuyên biệt.

3.2. XÁC ĐỊNH YÊU CẦU.

3.2.1. Nghiệp vụ

3.2.1.1. Đối tượng sử dụng

- Người mới bắt đầu làm quen với chỉnh sửa
- Người muốn cắt ghép video, nhưng gặp giới hạn về hiệu năng
- Người không chuyên, không có nhu cầu cao về chức năng chỉnh sửa

3.2.1.2. Chức năng

- Cắt video.
- Ghép video.
- Thay đổi kích thước khung ảnh.
- Thay đổi tỉ lệ khung ảnh.
- Thay đổi các thông số video
- Thay đổi tốc độ phát của video
- Đổi tên mở rộng cho video

3.3. ĐÁNH GIÁ VÀ LỰA CHỌN CÔNG NGHỆ

3.3.1. Mục tiêu đề tài

Xây dựng một website cho phép người dùng đăng tải các video lên và thực hiện các thao tác chỉnh sửa video như cắt ghép và chỉnh sửa các thông số cơ bản. Với mục tiêu là tận dụng toàn bộ giới hạn của web-design để tạo ra các thao tác tương đồng với các thao tác chuyên nghiệp.

3.3.2. Nội dung

Tìm hiểu lý thuyết và giới hạn về các công nghệ trong việc thực hiện đề tài:

- Tìm hiểu về NodeJS và Express và giới hạn tương tác giữa server và client
- Tìm hiểu về Angular về lý thuyết và giới hạn có thể thực hiện.
- Tìm hiểu về công cụ cho phép thực hiện can thiệp vào các tệp đa phương tiện: FFmpeg

Phân tích yêu cầu và hiện thực hóa chương trình:

- Thiết kế các lượt đồ
- Mô tả chức năng
- Thiết kế giao diện
- Xử lý dữ liệu tương tác giữa client và server

3.3.3. Môi trường lập trình

Về môi trường lập trình, cả front-end và back-end đều sử dụng phần mềm code editor Visual Studio Code để phát triển, tổ chức vào quản lý code.

Trong đó Visual Studio Code là một code editor miễn phí và phát triển trên đa nền tảng (Microsoft Windows, MacOS, Linux) được phát triển và phát hành bởi Microsoft. Code Editor này có thể coi như là một phiên bản nhỏ gọn và hợp xu hướng của Visual Studio. Có thể nói là code editor miễn phí có số lượng người dùng lớn nhất và được nhiều người nghĩ tới đầu tiên khi muốn coding.

3.3.4. Phương pháp nghiên cứu

Nghiên cứu về lý thuyết khi ứng dụng Angular và NodeJS làm cốt lõi của chương trình, từ đó đưa ra giới hạn hiện tại của bản thân và công nghệ sắp xếp lại các tính năng

Mô hình hóa và tham chiếu các chương trình đã có sẵn như DaVinci Resolve, capcut, Adobe Premiere từ đó sử dụng HTML5 triệt để nhằm giả lập lại các thao tác tương tự

Tham khảo ý kiến từ cộng đồng, giáo viên hướng dẫn, bạn bè cùng với kinh nghiệm lập trình bản thân hấp thụ được trong thời gian học.

3.3.5. Ý nghĩa khoa học, thực tiễn

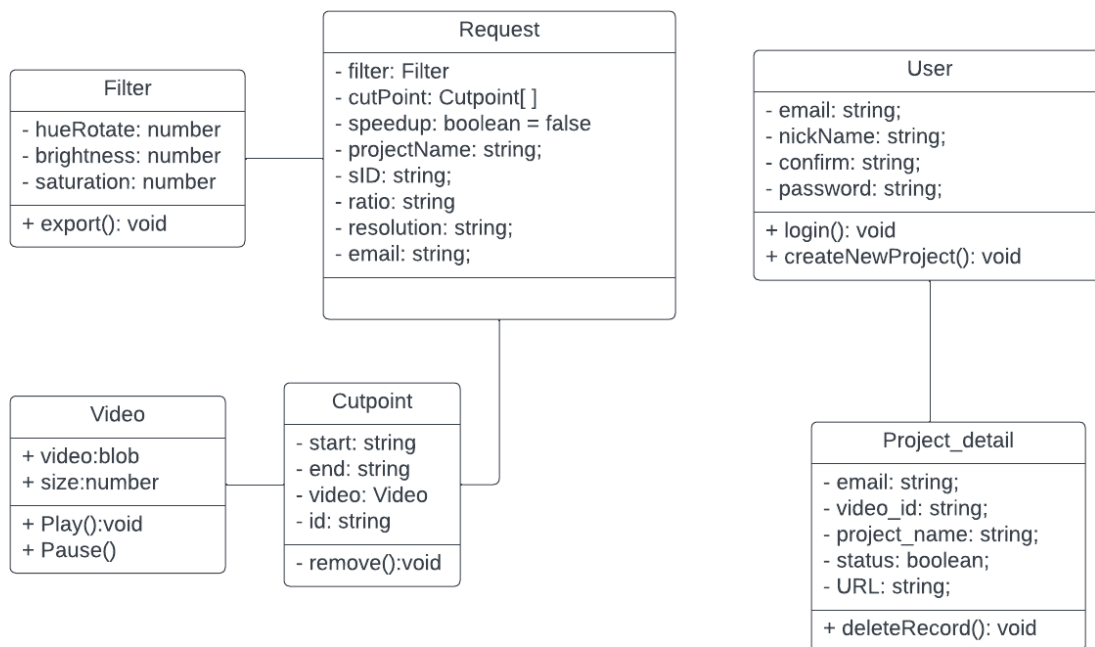
Về khoa học, chương trình có thể tận dụng tất cả khả năng của HTML5, CSS, JavaScript để tạo nên nền giao diện thao tác chuyên nghiệp. Từ đó nghiên cứu được khả năng đích thực trong thời điểm này

Về thực tiễn, nghiên cứu được các thao tác cơ bản của người dùng từ đó tìm hiểu và điều chỉnh giao diện phù hợp với thói quen thao tác.

CHƯƠNG 4. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.

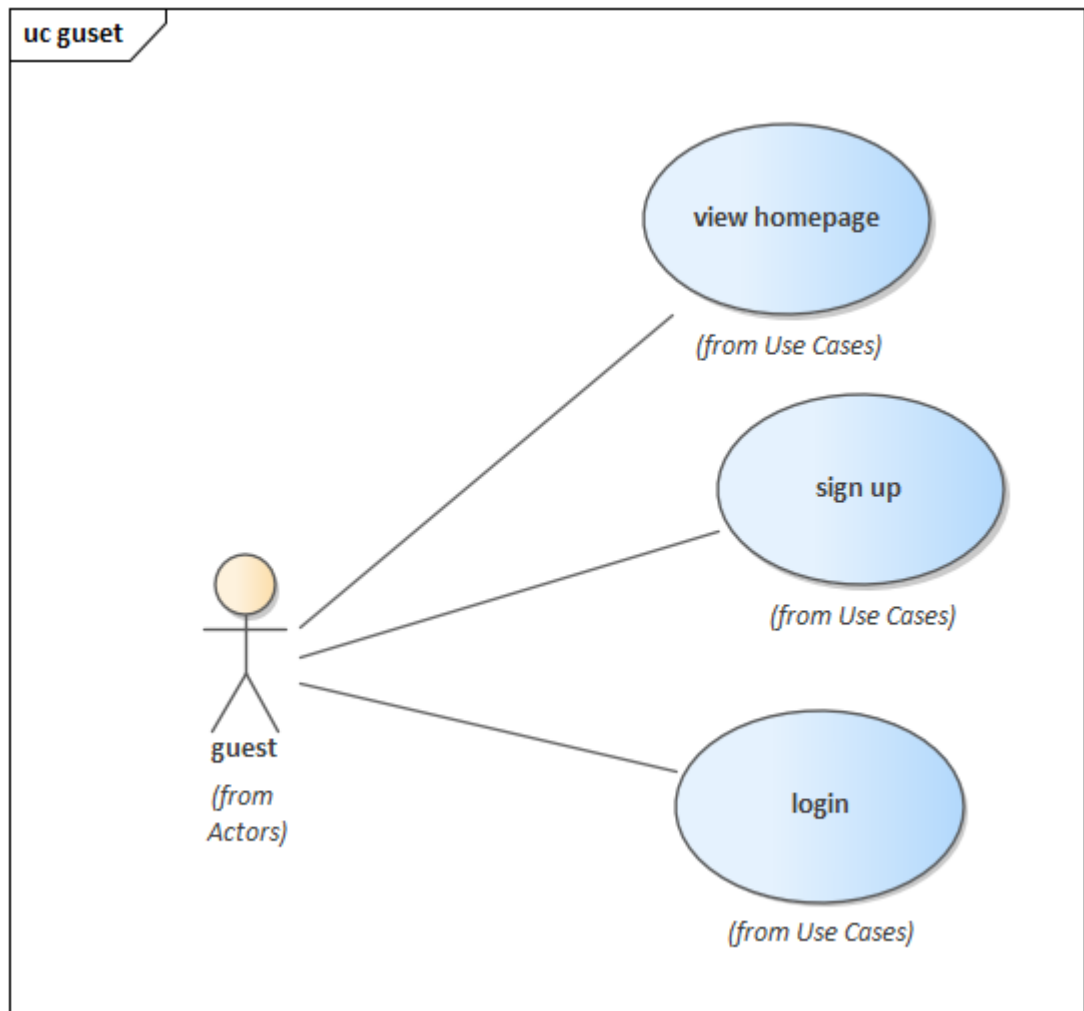
4.1. PHÂN TÍCH HỆ THỐNG

4.1.1. Sơ đồ lớp.

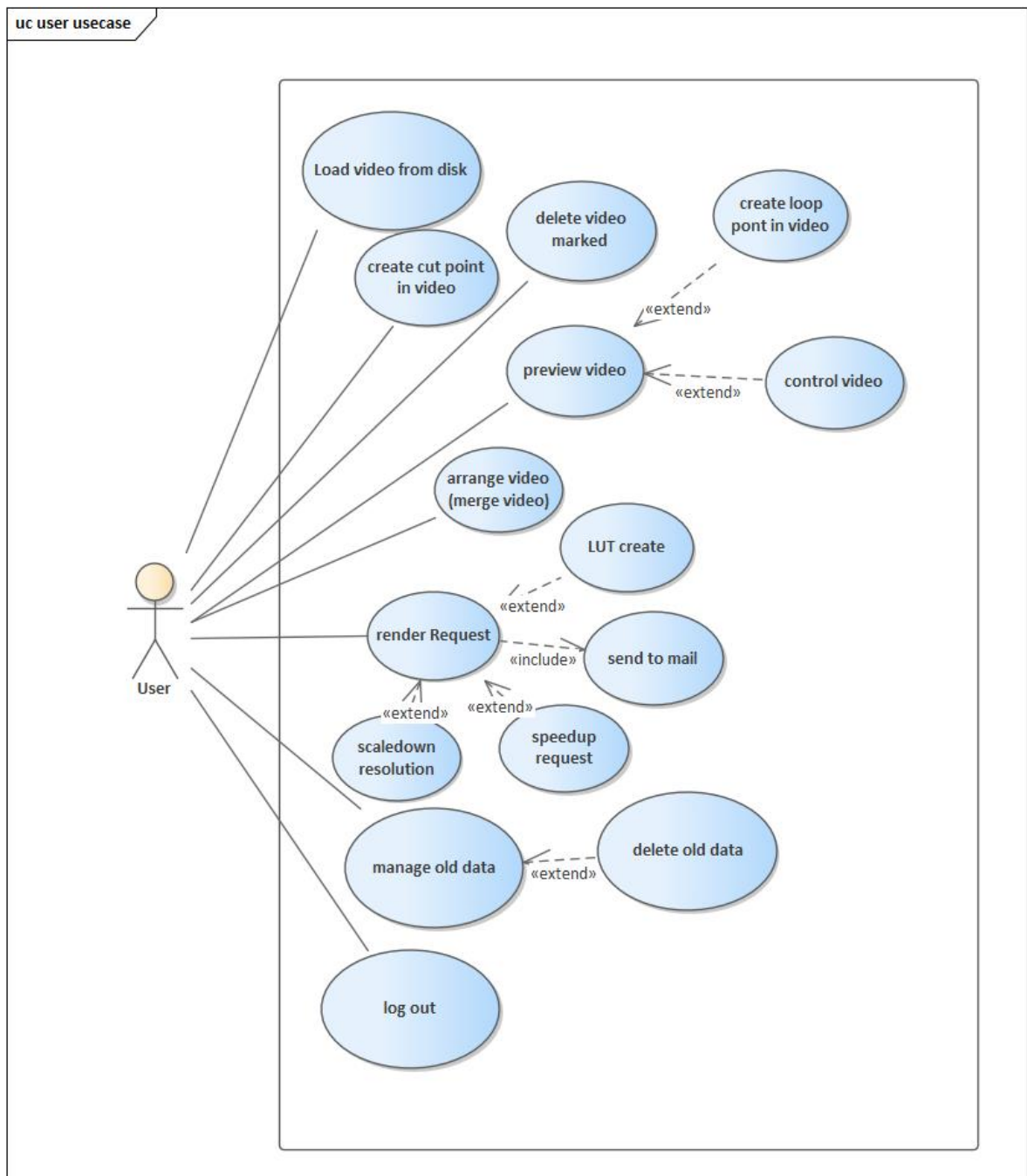


Hình 4 Hình ảnh về sơ đồ lớp

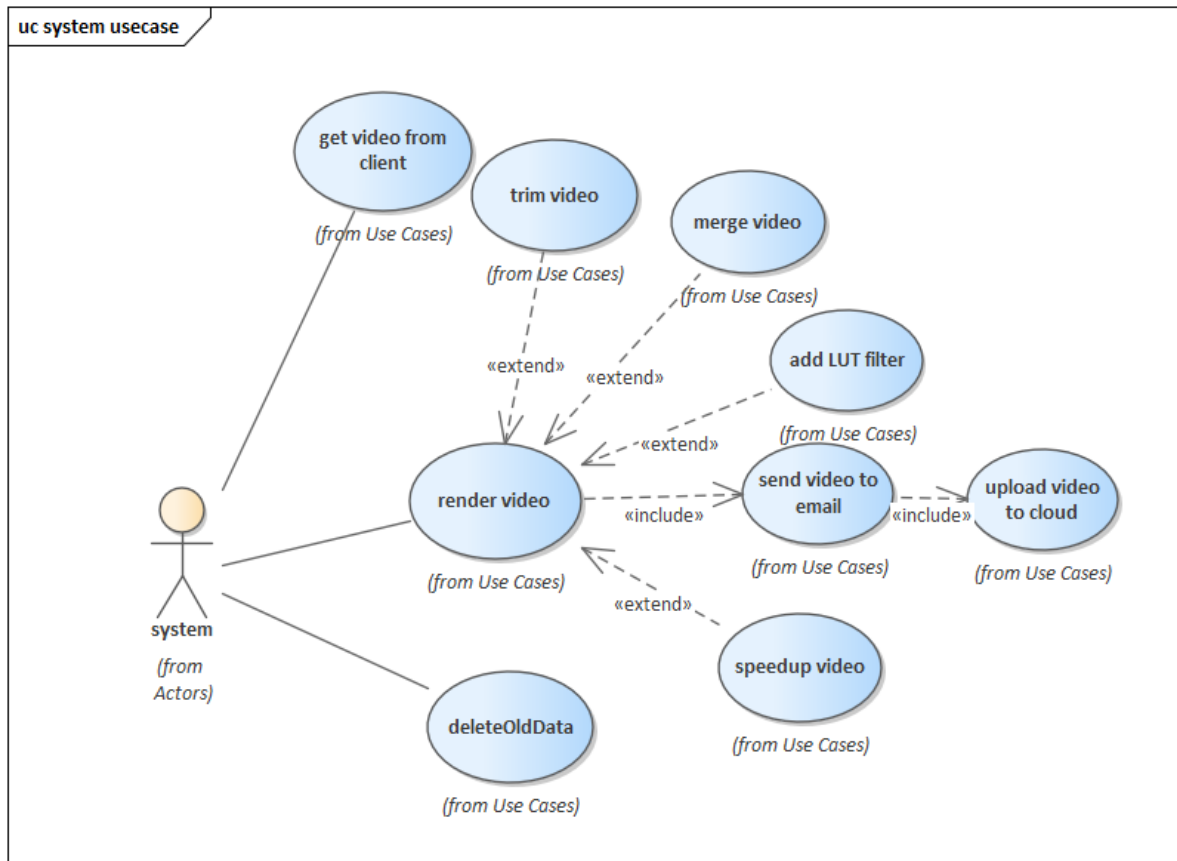
4.1.2. Sơ đồ usecase



Hình 5 Hình ảnh Usecase của khách



Hình 6 Hình ảnh Usecase của người dùng



Hình 7 Hình ảnh usecase của máy

4.1.3. Mô tả sơ đồ usecase

Bảng 3: Bảng mô tả nội chi tiết dung sơ đồ usecase

STT	Chức năng	Mô tả trên hình	Mô tả chi tiết
UC_1	Đăng ký	Sign up	Là chức năng khách có thể đăng ký trở thành User của hệ thống và được truy cập vào chương trình chỉnh sửa
UC_2	Đăng nhập	Login	Là chức năng giúp người dùng từ khách đăng nhập vào hệ thống và sử dụng tiện ích của chương trình
UC_3	Đăng tải video	Load video from disk	Load video từ ổ đĩa lên trên trang web, đồng thời gửi video lên trên server
UC_4	Tạo ra các điểm cắt video	Create cut point in video	Tạo ra các điểm cắt từ đó có thể đưa ra thông số cho server render
UC_5	Xóa các điểm đánh dấu	Delete video marked	Xóa các điểm đánh dấu các đoạn cắt
UC_6	Xem lại các video đã upload lên	Preview video	Xem lại các video đã chọn, từ đó dễ dàng đưa ra quyết định về các điểm cắt của video sau cùng, và xem lại kết quả cuối cùng
UC_7	Tạo vị trí lặp lại	Create loop point in video	Tạo ra vị trí lặp lại trên video, khi phát video thì video sẽ chỉ lặp lại đoạn được đánh dấu
UC_8	Điều khiển video	Control video	Điều khiển video phát, dừng, hoặc lặp đi lặp lại, ...
UC_9	Sắp xếp lại các video	Arrange video	Sắp xếp lại các đoạn cắt từ đó đưa ra đoạn video hoàn chỉnh, thứ tự của các video được sắp xếp là thứ tự của các video sau khi xếp lại

UC_10	Tạo màn hình lọc ảnh	LUT create	Tạo ra lớp màng giả lập tại client để người nhìn có thể thay đổi nhóm màu sắc trên của video thông qua vòng tròn HUE
UC_11	Yêu cầu tăng tốc độ video	Speedup request	Yêu cầu server render ra video có tốc độ tăng lên so với tốc độ đầu vào của video
UC_12	Giảm độ phân giải	Scaledown resolution	Yêu cầu server render ra video có độ phân giải giảm nhỏ hơn so với ban đầu
UC_13	Gửi yêu cầu render	Render request	Yêu cầu server render video theo mô tả video được sắp xếp, trong đó gồm các thông số thuộc tính của video
UC_14	Đăng xuất	Log out	Là chức năng giúp người dùng thoát khỏi hệ thống
UC_15	Quản lý video	Manage video data	Là chức năng để người dùng quản lý được các video project đã trả dữ liệu trước đó đã thực hiện.
UC_16	Cắt video	Trim video	Cắt các video theo yêu cầu của người dùng, được đánh dấu bằng điểm bắt đầu và kết thúc
UC_17	Ghép video	Merge video	Nối các video lại theo thứ tự đã được đề xuất từ yêu cầu render từ client
UC_18	Nhận video từ client	Get video from client	Hoạt động chung UC_1, chức năng sẽ nhận video và trả về tên video được đánh dấu, đồng thời là nhận dạng cho lượt hoạt động
UC_19	Tăng tốc video	Speed up video	Tăng tốc video lên theo yêu cầu từ render request

UC_20	Render video	Render video	Render video theo yêu cầu được gửi từ client, thông qua các thông số gửi kèm
UC_21	Xóa dữ liệu cũ	deleteOldData	Xóa các dữ liệu cũ tồn đọng trên server
UC_22	Tạo lớp màng lên video	Add LUT filter	Server tạo ra một lớp lọc để thay đổi màu sắc trên video theo hệ vòng tròn HUE
UC_23	Đăng tải video lên cloud	Upload video to cloud	Hệ thống sẽ đưa video lên cloud để lưu trữ lại video với mục đích làm nguồn dẫn khi gửi video thông qua mail
UC_24	Gửi video qua mail	Send video to mail	Video sau khi được render xong, server sẽ gửi video kết quả lên cloud để lưu trữ lại và gửi video này thông qua mail để đảm bảo dung lượng mail khi gửi.

4.1.4. Đặc tả Usecase**4.1.4.1. Usecase đăng ký****Bảng 4 Bảng mô tả chi tiết usecase đăng ký**

Usecase đăng ký	
Mô tả	Là chức năng giúp khách có thể đăng ký trở thành người dùng
Tác nhân kích hoạt	Guest
Tiền điều kiện	Không
Các bước thực hiện	(1) Bấm vào nút Sign up. (2) Nhập email và nickname (3) Nhấn đăng ký (4) Kiểm tra email đăng ký để hoàn thành xác thực
Thành công	Hiện thị nội dung xác thực thành công.
Thất bại	Hiện thị nội dung đăng ký thất bại

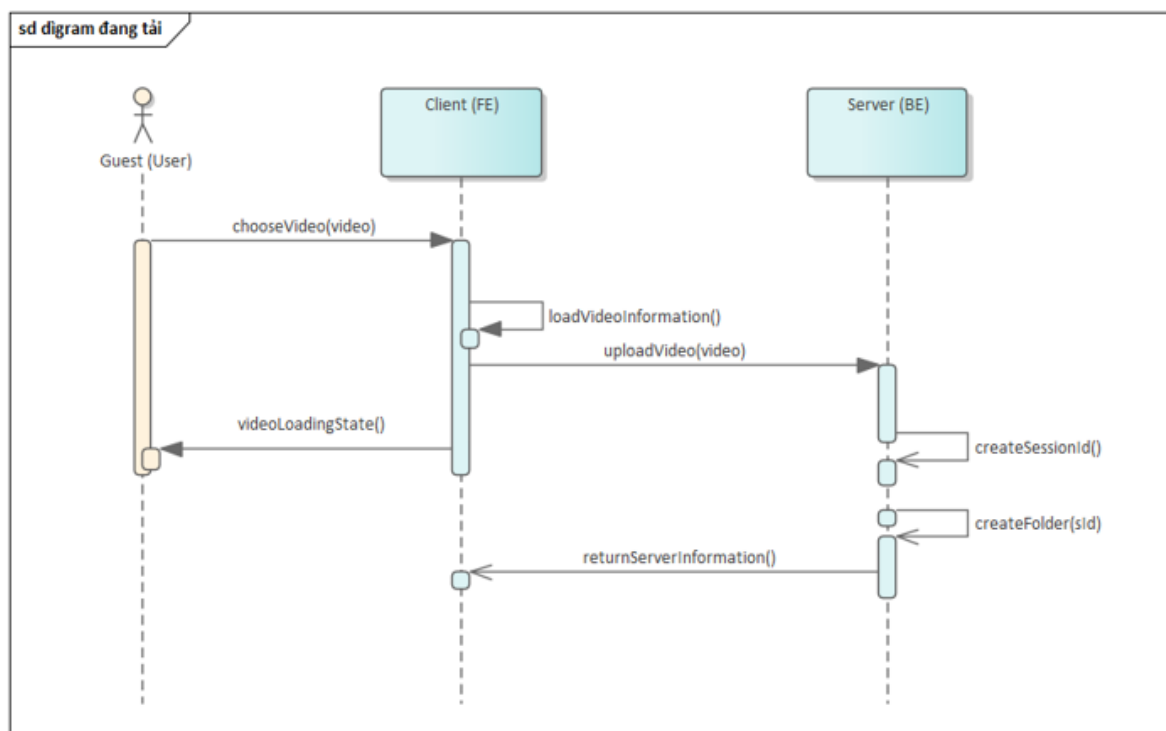
4.1.4.2. Usecase đăng nhập**Bảng 5 Bảng mô tả chi tiết usecase đăng nhập**

Usecase đăng nhập	
Mô tả	Đây là chức năng để người dùng đăng nhập vào hệ thống
Tác nhân kích hoạt	Guest
Tiền điều kiện	Không
Các bước thực hiện	(1) Bấm vào nút login (2) Nhập email. (3) Nhận mã OTP từ email. (4) Nhập OTP. (5) Bấm nút đăng nhập
Thành công	Chương trình sẽ chuyển hướng về trang quản lý nội dung
Thất bại	Thông báo lỗi

4.1.4.3. Usecase đăng tải video

Bảng 6 Bảng mô tả chi tiết usecase đăng tải video

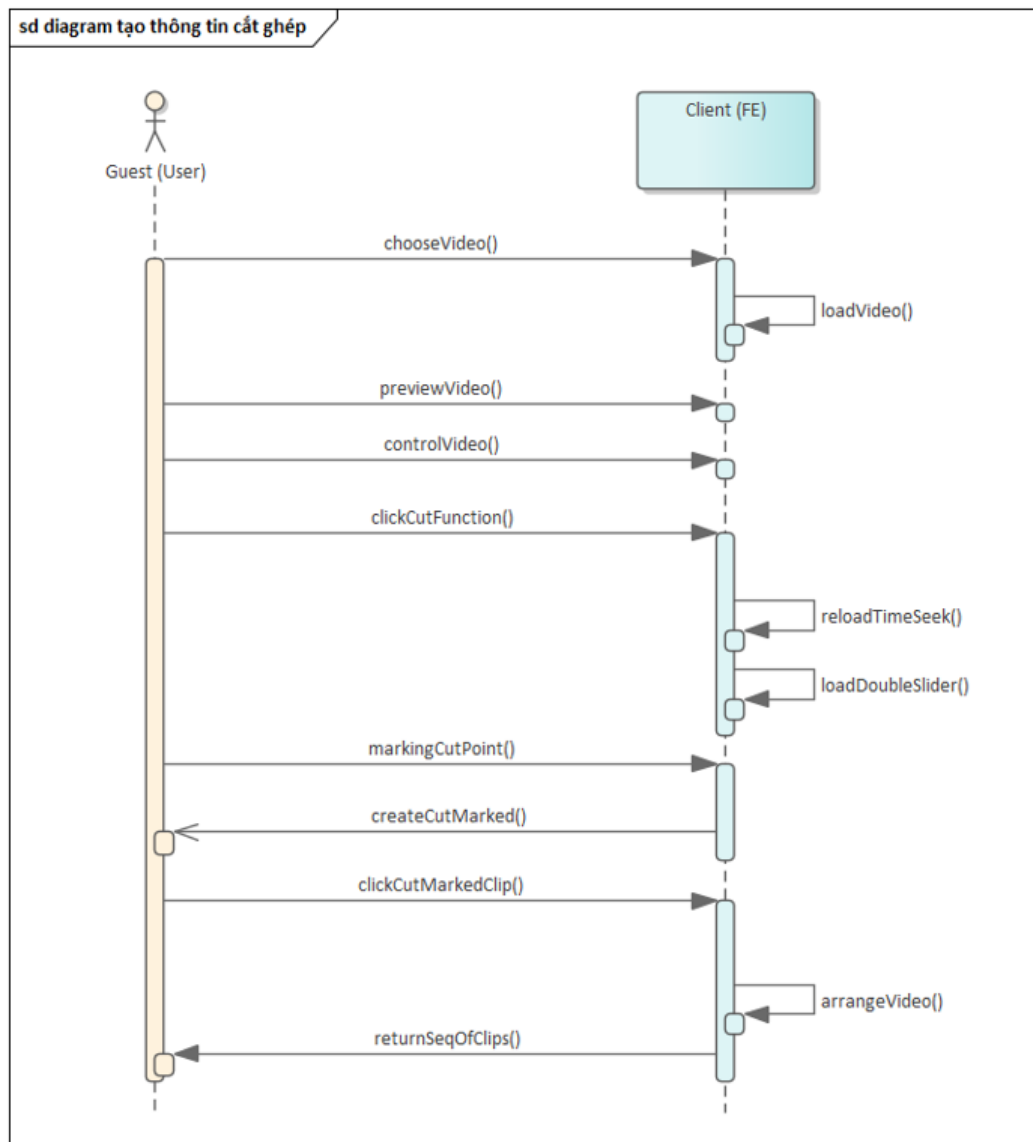
Usecase đăng tải video	
Mô tả	Đây là chức năng khi người dùng load một video lên chương trình từ đó chương trình tạo ra một bản ghi tại client song song đó gửi đoạn video lên server
Tác nhân kích hoạt	User
Tiền điều kiện	Hoàn thành đăng nhập
Các bước thực hiện	(1) Bấm vào nút upload video (2) Chọn các video cần thao tác (3) Nhấn đăng tải
Thành công	Chương trình cập nhật đoạn video lên giao diện đăng tải
Thất bại	Không thể hiện nội dung bị lỗi lên giao diện

**Hình 8** Hình ảnh lược đồ tuần tự chức năng đăng tải video

4.1.4.4. Usecase tạo điểm cắt video

Bảng 7 Bảng mô tả chi tiết usecase tạo điểm cắt video

Usecase tạo điểm cắt video	
Mô tả	Chức năng hỗ trợ người dùng tạo ra các điểm cắt của từng đoạn video, tạo nên các đánh dấu, tạo cảm giác như cắt các đoạn cắt
Tác nhân kích hoạt	User
Tiền điều kiện	Có video được đăng tải thành công lên giao diện
Các bước thực hiện	(1) Chọn một video đã được đăng tải lên giao diện (2) Chọn vào chức năng tạo các điểm cắt video (3) Chọn ra hai điểm đánh dấu đầu vào cuối của video (4) Bấm chọn tạo đánh dấu
Thành công	Điểm đánh dấu sẽ xuất hiện tại giao diện danh sách điểm cắt
Thất bại	Không thể hiện đoạn cắt tại giao diện danh sách điểm cắt



Hình 9 Hình ảnh lược đồ tuần tự tạo thông tin cắt ghép

4.1.4.5. Usecase gửi yêu cầu render.

Bảng 8 Bảng mô tả chi tiết usecase yêu cầu render.

Usecase gửi yêu cầu render video	
Mô tả	Chức năng thao tác tại client giúp người dùng gửi yêu cầu render video lên server từ đó server trực tiếp render ra video theo yêu cầu
Tác nhân kích hoạt	User
Tiền điều kiện	Có video tại giao diện sắp xếp vị trí video
Các bước thực hiện	(1) Bấm chọn vào nút yêu cầu render. (2) Chọn khung hình cho video (3) Chọn tốc độ cho video (4) Bấm chọn gửi yêu cầu
Thành công	Màn hình loading khởi động, sau khi video đã được chuẩn bị hoàn tất, video sẽ được gửi trực tiếp về máy theo phương thức tải xuống

4.1.4.6. Usecase nhận video từ client

Bảng 9 Bảng mô tả chi tiết usecase nhận video từ client

Usecase nhập video từ client	
Mô tả	Video được nhận từ client, đánh dấu lại video vừa được tải lên.
Tác nhân kích hoạt	Server
Tiền điều kiện	Có video được gửi từ client
Các bước thực hiện	(1) Bắt video được gửi từ client (2) Tạo ra unique session id (3) Tạo folder cho video được gửi lên theo session id (4) Copy video vào thư mục vừa tạo (5) Gửi lại thông tin về unique id về client, để nhận dạng đánh dấu
Thành công	Loading icon tại client sẽ được tắt đi

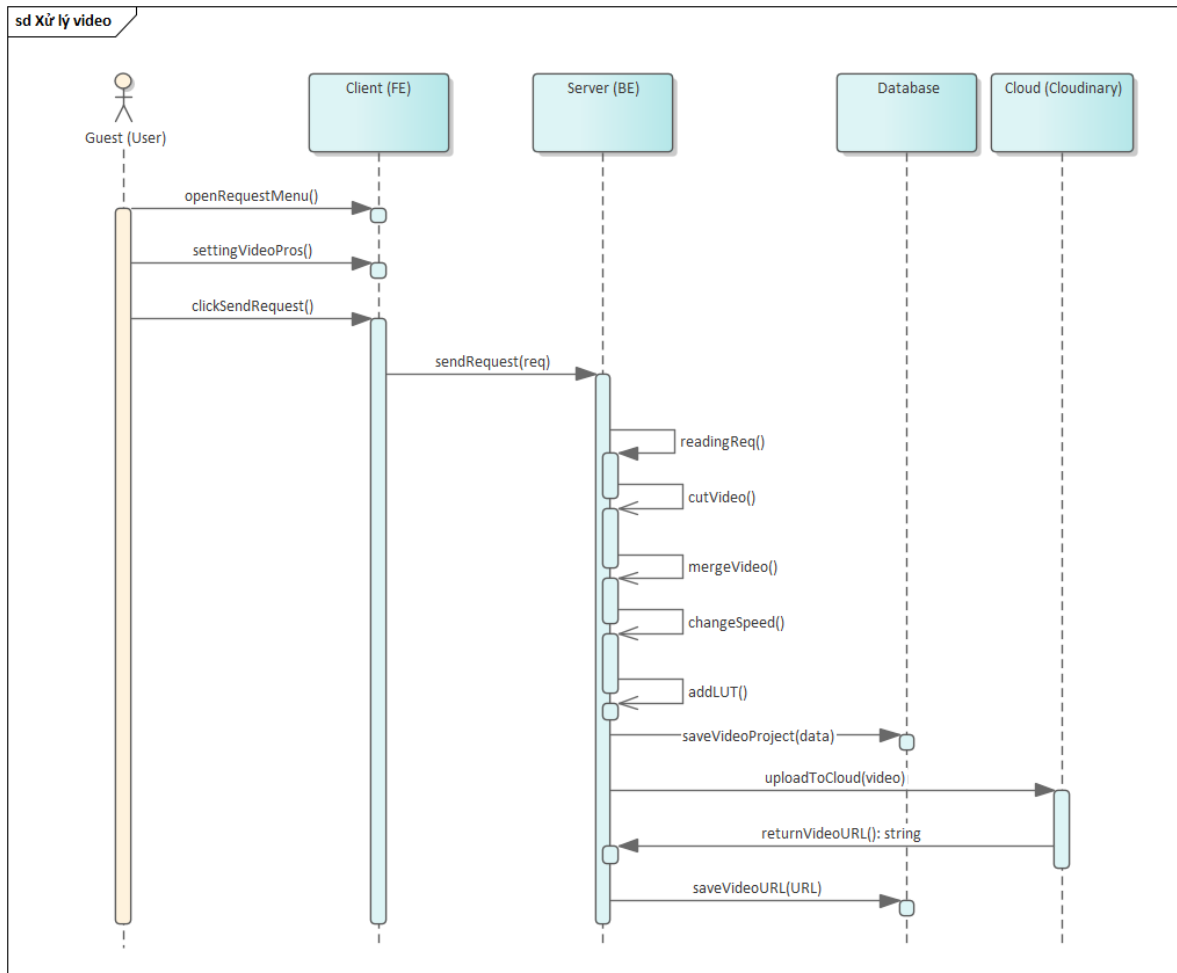
4.1.4.7. Usecase ghép video

Bảng 10 Bảng mô tả chi tiết usecase ghép video

Usecase ghép video	
Mô tả	Chức năng ghép video theo từng đôi một
Tác nhân kích hoạt	Server
Tiền điều kiện	Thông tin trong yêu cầu cắt có ít nhất hai dữ kiện
Các bước thực hiện	(1) Nhận thông tin từ các yêu cầu (2) Lấy thông tin về hai video cần ghép lại (3) Tạo ra tên giả cho video được ghép lại (4) Xác định yêu cầu sau khi ghép hai video lại (5) Thực hiện ghép hai video theo yêu cầu (6) Gửi lại tên giả của video lại cho vị trí yêu cầu
Thành công	Chương trình cập nhật đoạn video lên giao diện đăng tải
Thất bại	Không thể hiện nội dung bị lỗi lên giao diện

4.1.4.8. *Usecase render video.***Bảng 11** Bảng mô tả chi tiết usecase render video.

Usecase render video	
Mô tả	Server nhận yêu cầu từ client, dựa vào các thông tin đặt ra
Tác nhân kích hoạt	Server
Tiền điều kiện	Nhận được yêu cầu render từ client
Các bước thực hiện	(1) Nhận thông tin từ yêu cầu (2) Phân tích thông tin từ yêu cầu (3) Tạo ra các đoạn cắt như yêu cầu đề ra (4) Ghép nối video theo từng đôi một theo yêu cầu (5) Xử lý các yêu cầu phụ từ client (6) Gửi video lên dịch vụ đám mây (7) Gửi email cho người dùng về thông tin lấy từ đám mây
Thành công	Email người dùng sẽ nhận được mail về thông tin của video được đăng tải lên dịch vụ đám mây.



Hình 10 Hình ảnh lược đồ tuần tự xử lý video

4.1.4.9. *Usecase addLUTFilter***Bảng 12** Bảng mô tả chi tiết usecase addLUTFilter

Usecase addLUTFilter	
Mô tả	Đây là chức năng để cho server render bộ màu sắc cho video
Tác nhân kích hoạt	Server
Tiền điều kiện	Có được tên của giai đoạn trước đó thực hiện
Các bước thực hiện	(1) Kiểm tra thông tin từ request (2) Phân tích thông tin từ yêu cầu (3) Sắp xếp các thuộc tính lại. (4) Thực hiện việc thêm bộ lọc màu (5) Gửi lại tên giả của video tại vị trí yêu cầu
Thành công	Dữ liệu về tên của video kết quả

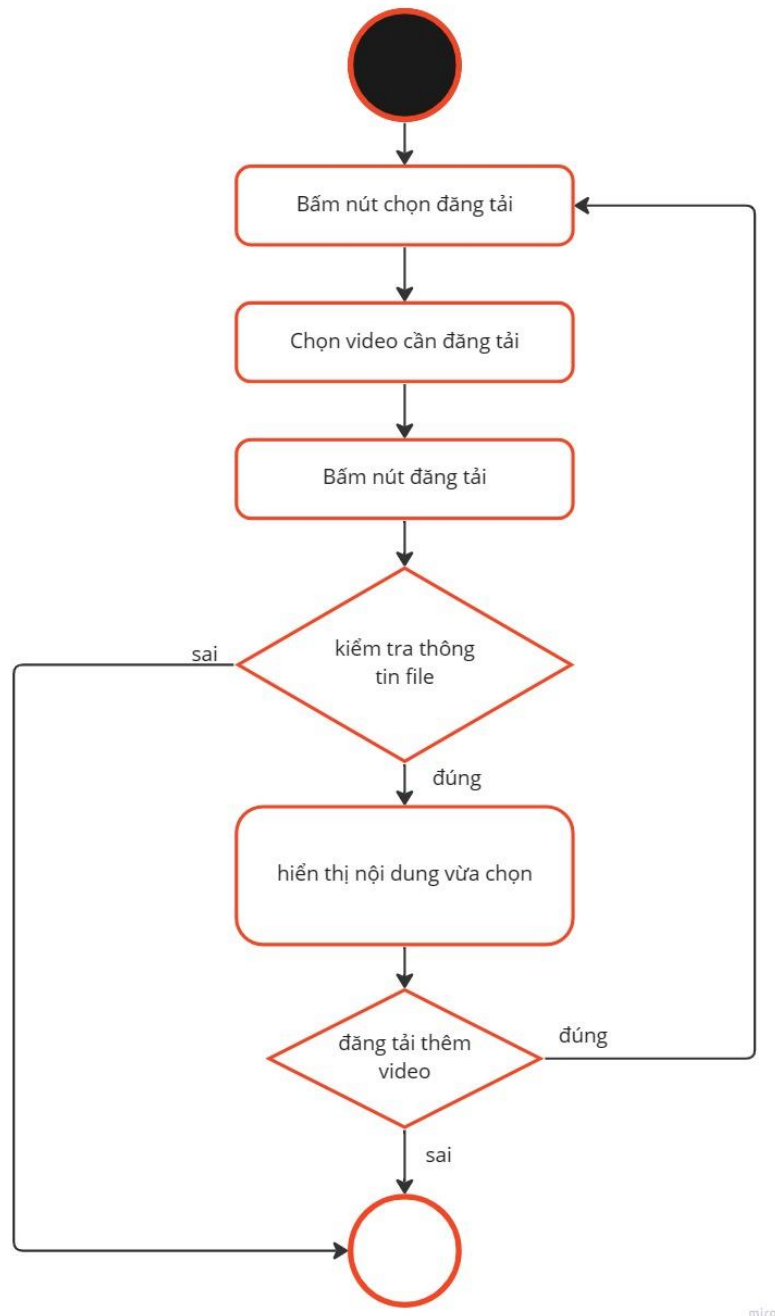
4.1.4.10. *Usecase send video to e-mail.***Bảng 13** Bảng mô tả chi tiết usecase send video to email.

Usecase send video to email	
Mô tả	Đây là chức năng để cho server sau khi render sẽ gửi video đã hoàn thành vào mail của người dùng
Tác nhân kích hoạt	Server
Tiền điều kiện	Video kết quả được render xong
Các bước thực hiện	(1) Kiểm tra thông tin từ request (2) Lấy email từ yêu cầu. (3) Kiểm tra thông tin của video kết quả (4) Gửi video kết quả lên cloud và nhận về liên kết (5) Lấy địa chỉ liên kết đưa vào nội dung mail (6) Thiết lập email gửi (7) Gửi email.
Thành công	Người dùng nhận được email từ hệ thống, trong đó có liên kết tới video thực hiện

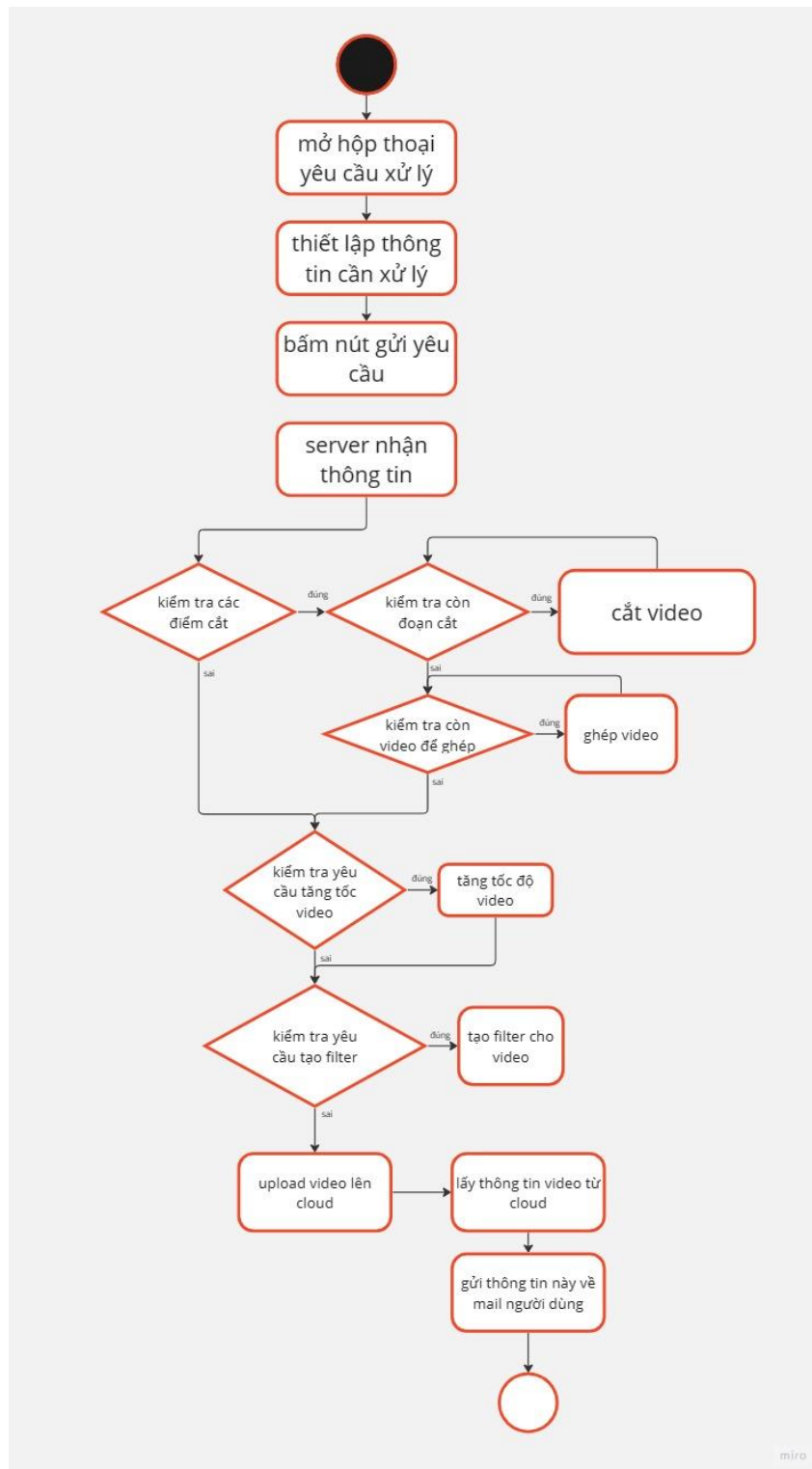
4.1.4.11. Usecase xóa dữ liệu cũ**Bảng 14** Bảng mô tả chi tiết usecase xóa dữ liệu cũ

Usecase xóa dữ liệu cũ	
Mô tả	Đây là chức năng để cho server giảm bớt dung lượng trên server
Tác nhân kích hoạt	Server
Tiền điều kiện	Không
Các bước thực hiện	(1) Kiểm tra thời gian hoạt động chương trình (2) Load toàn bộ thông tin của các session folder (3) Kiểm tra thời gian chỉnh sửa của SFolder (4) Xóa folder có thời gian tồn đọng trên 7 ngày
Thành công	Dữ liệu về các folder
Thất bại	Không biểu hiện

4.2. THIẾT KẾ HỆ THỐNG

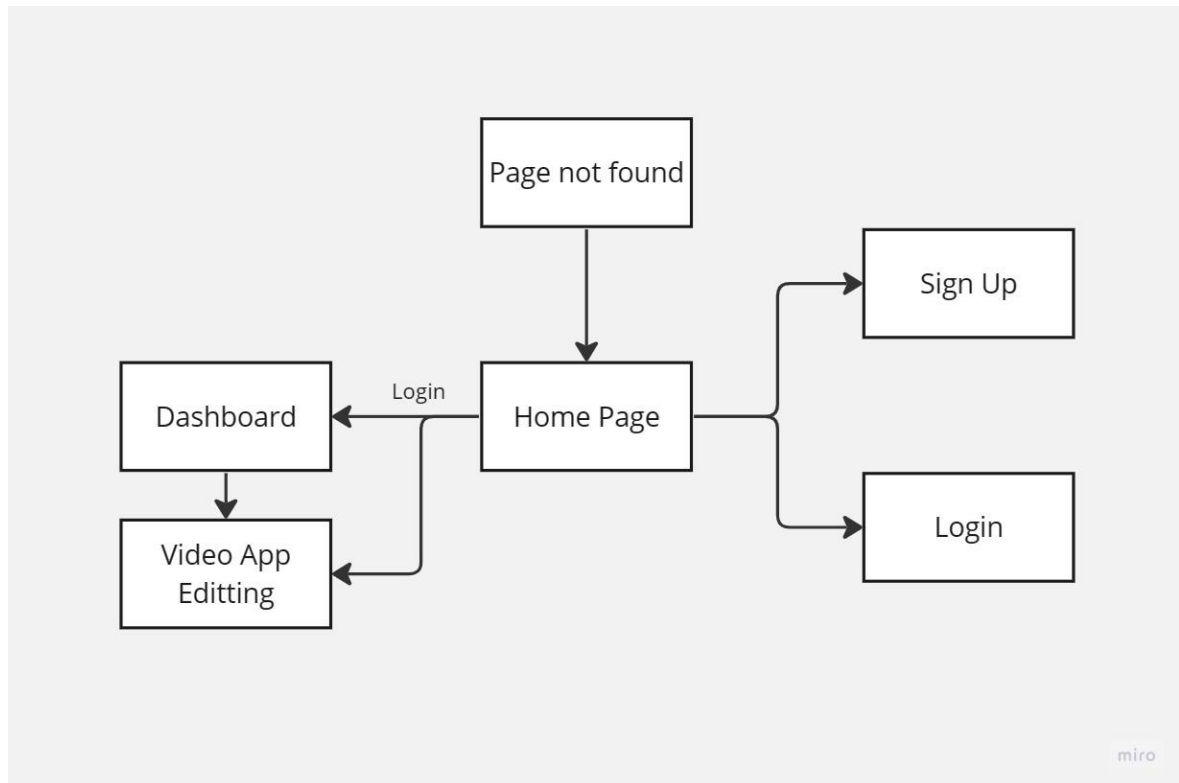


Hình 11 Hình ảnh lược đồ hoạt động của chức năng đăng tải video



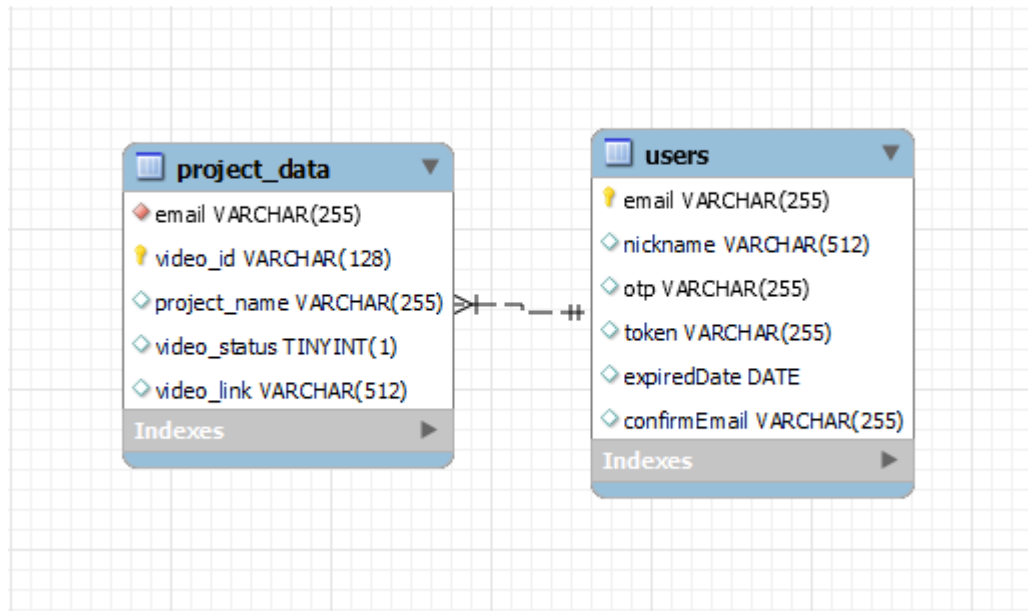
Hình 12 Hình ảnh lược đồ hoạt động của chức năng xử lý video

4.2.1. Tổng quát chương trình



Hình 13 Hình ảnh thể hiện cách chuyển trang trong chương trình

4.2.2. Tổng quan về Database



Hình 14 Hình ảnh sơ đồ ER của database

Bảng 15 Bảng thể hiện chi tiết bảng Users

Bảng User mục đích lưu trữ các thông tin về người dùng và quy tắc đăng nhập.			
STT	Thuộc tính	Kiểu dữ liệu	Mục đích
1	Email	Varchar (Primary key)	Là khóa chính, là thông tin độc nhất trong bảng, mỗi email được coi như một tài khoản được đăng ký vào hệ thống chương trình
2	Nickname	Varchar	Là trường để lưu lại tên người dùng.
3	OTP	Varchar	Là trường lưu lại mật khẩu, có độ dài lớn, để mật khẩu sau khi bấm ra có thể chứa được. và được xóa sau khi hoàn thành đăng nhập.
4	Token	Varchar	Là trường lưu token của người dùng
5	expiredDate	Date	Là trường thể hiện thời gian hết hạn đăng nhập của người dùng

6	confirmEmail	Varchar	Là trường lưu thông tin xác thực khi đăng ký. Sau khi đăng ký hoàn tất thì dữ liệu sẽ thay đổi thể hiện hoàn thành.
---	--------------	---------	---

Bảng 16 Bảng thể hiện chi tiết bảng Project_data

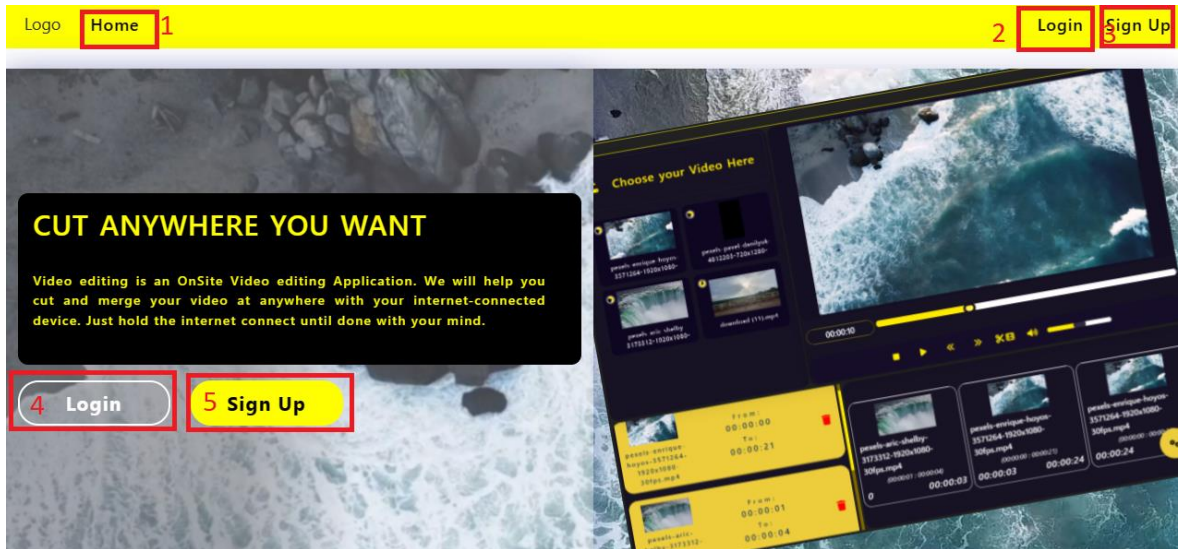
Bảng Project_data nhằm mục đích lưu trữ các thông tin liên quan tới các video mà người dùng đã yêu cầu hệ thống thực hiện.			
STT	Thuộc tính	Kiểu dữ liệu	Mục đích
1	<u>email</u>	Varchar (Foreign Key)	Là khóa ngoại, tham chiếu tới email tại bảng users, nhằm xác định chính xác email đã từng đăng ký
2	<u>video_id</u>	Varchar (Primary key)	Được thiết lập là khóa chính, mỗi id là duy nhất để xác định
3	project_name	Varchar	Là tên của lần chỉnh sửa video, người dùng sẽ tạo ra một tên gợi nhớ và lưu lại
4	video_status	Tinyint	Là dữ liệu để biết được video đó đang được xử lý hoặc đã hoàn thành
5	video_link	Varchar	Là trường dữ liệu thể hiện đường dẫn với video được lưu tại đám mây, đường dẫn này sẽ xuất hiệu sau khi video_status được chuyển thành trạng thái hoàn thành.

4.3. THIẾT KẾ GIAO DIỆN

4.3.1. Danh sách giao diện người dùng.

4.3.1.1. Mô tả chi tiết giao diện

a. Giao diện trang chủ



Hình 15 Hình ảnh giao diện trang chủ

Bảng 17 Bảng chi tiết giao diện trang chủ

STT	Tên đối tượng	Loại	Mô tả
1	Home	Link	Là đường dẫn đưa ta trở về Homepage
2	Nav Login	Link	Là đường dẫn đưa ta tới trang đăng nhập
3	Nav Sign Up	Link	Là đường dẫn đưa ta tới trang đăng ký
4	Login Button	Button	Là nút để chuyển hướng tới trang đăng nhập
5	Sign Up Button	Button	Là nút để chuyển hướng tới trang đăng ký thành viên

b. Giao diện đăng nhập

The image displays two screenshots of a login interface with numbered annotations:

- 1**: Points to the "Email" label.
- 2**: Points to the email input field containing "tranvietmar15@gmail.com".
- 3**: Points to the "Get Password" button.
- 4**: Points to the "Password" label.
- 5**: Points to the password input field.
- 6**: Points to the "Login" button.

Hình 16 Hình ảnh giao diện login

Bảng 18 Bảng chi tiết giao diện login

STT	Tên đối tượng	Loại	Mô tả
1	Email label	Label	Là label thể hiện vị trí của email
2	Email input	Input[type=email]	Là vị trí tại đây nhập email để có thể lấy password
3	Get password	Button	Nút nhấn để yêu cầu hệ thống cung cấp mật khẩu
4	Password label	Label	Là label thể hiện vị trí của password
5	Password input	Input[type=password]	Là vị trí nhập liệu, tại đây nhập password để đăng nhập
6	login	button	Là nút nhấn để đăng nhập

c. Giao diện đăng ký

The image shows a registration form with the following elements:

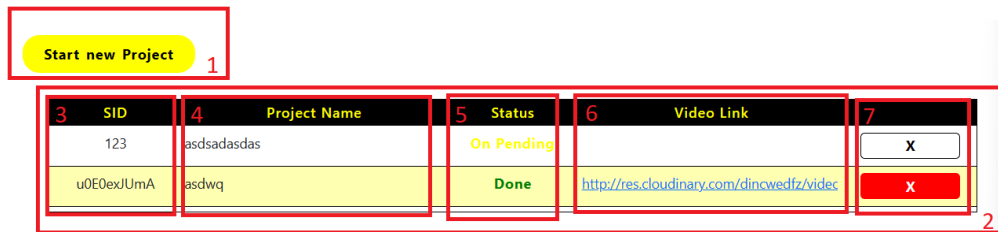
- 1**: Email label
- 2**: Email input field containing "18110238@student.hcmute.edu.vn"
- 3**: Nickname label
- 4**: Nickname input field containing "Nguyen Nguyen"
- 5**: Sign up button
- 6**: A message box stating: "Plese check your email to confirm your account. We will re-direct to homepage in now, or [click here](#)."

Hình 17 Hình ảnh giao diện đăng ký

Bảng 19 Bảng chi tiết giao diện đăng ký

STT	Tên đối tượng	Loại	Mô tả
1	Email label	Label	Là label để thể hiện vị trí nhập email
2	Email input	Input[type=email]	Là vị trí để nhập email muốn đăng ký trở thành thành viên
3	Nickname label	Label	Là label để thể hiện vị trí nhập nickname
4	Nickname input	Input	Là vị trí để nhập nickname
5	Sign up	Button	Là nút để gửi yêu cầu đăng ký thành viên
6	Reminder	Div	Là thể diện nhắc nhở việc tiếp theo của người dùng

d. Giao diện quản lý video

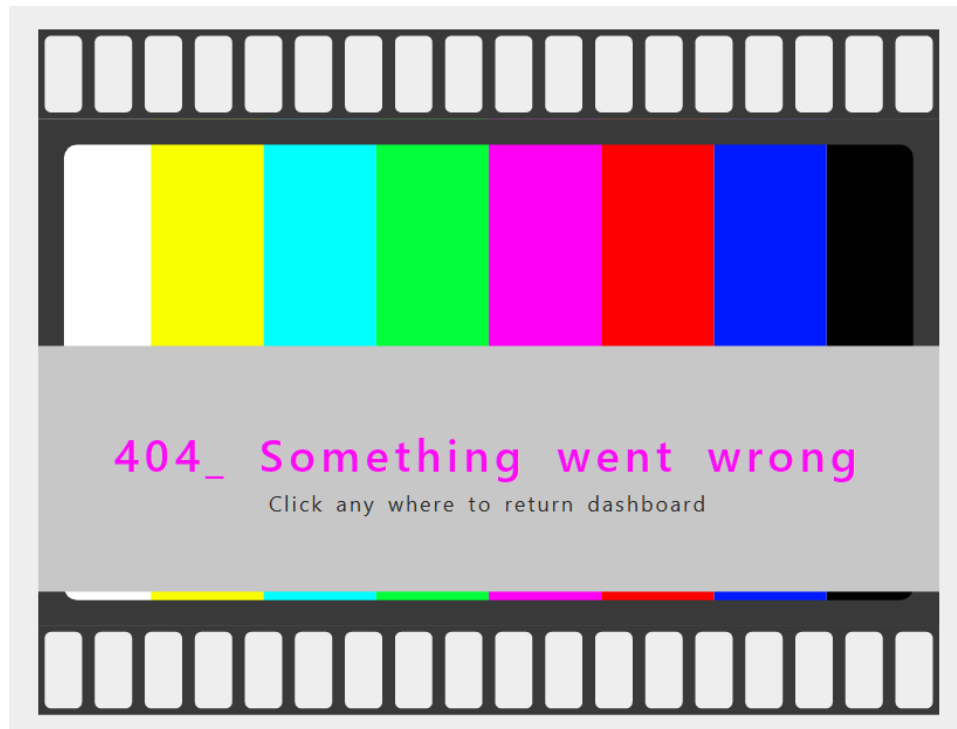


Hình 18 Hình ảnh giao diện dashboard

Bảng 20 Bảng chi tiết giao diện dashboard

STT	Tên đối tượng	Loại	Mô tả
1	Start new project	Button	Là nút để tạo ra nội dung mới
2	Khung quản lý	Div	Là khung bao quanh lại nội dung của thông tin
3	video id	String	Thể hiện video id
4	project name	String	Thể hiện tên được người dùng đặt để gọi nhớ
5	status	boolean	Thể hiện trạng thái của video
6	video link	Link	Thể hiện đường dẫn tới video sau khi được upload lên cloud
7	chức năng	Button	Là vị trí đặt nút xóa, nó sẽ xóa đi bản ghi theo video id

e. Giao diện pagenotfound



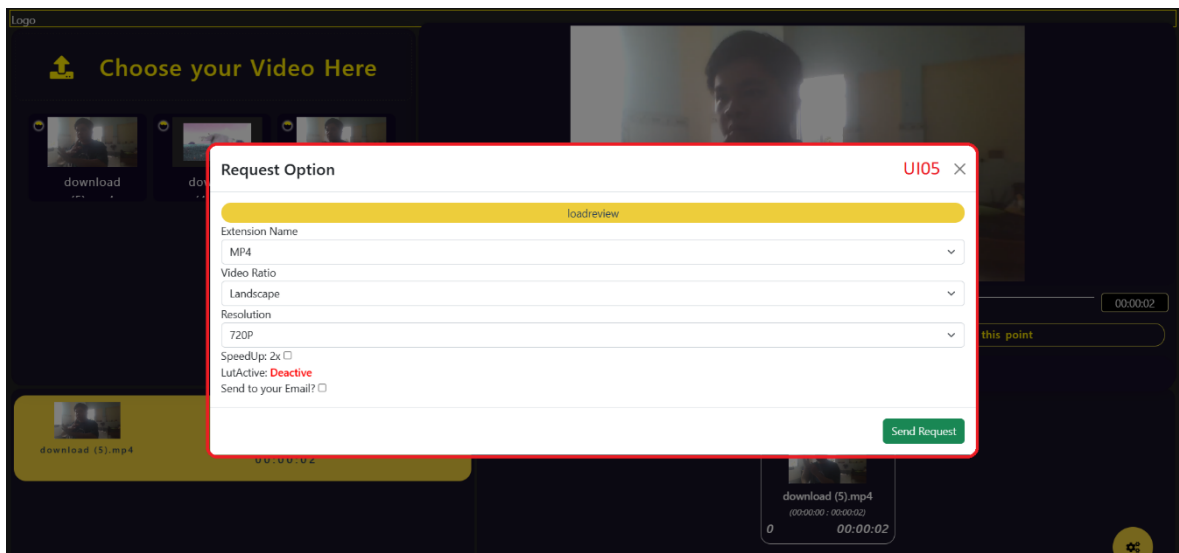
Hình 19 Hình ảnh giao diện Page Not Found

4.3.2. Danh sách giao diện chương trình chính

4.3.2.1. Giao diện chung



Hình 20 Hình ảnh giao diện chung của chương trình



Hình 21 Hình ảnh giao diện chung của chương trình

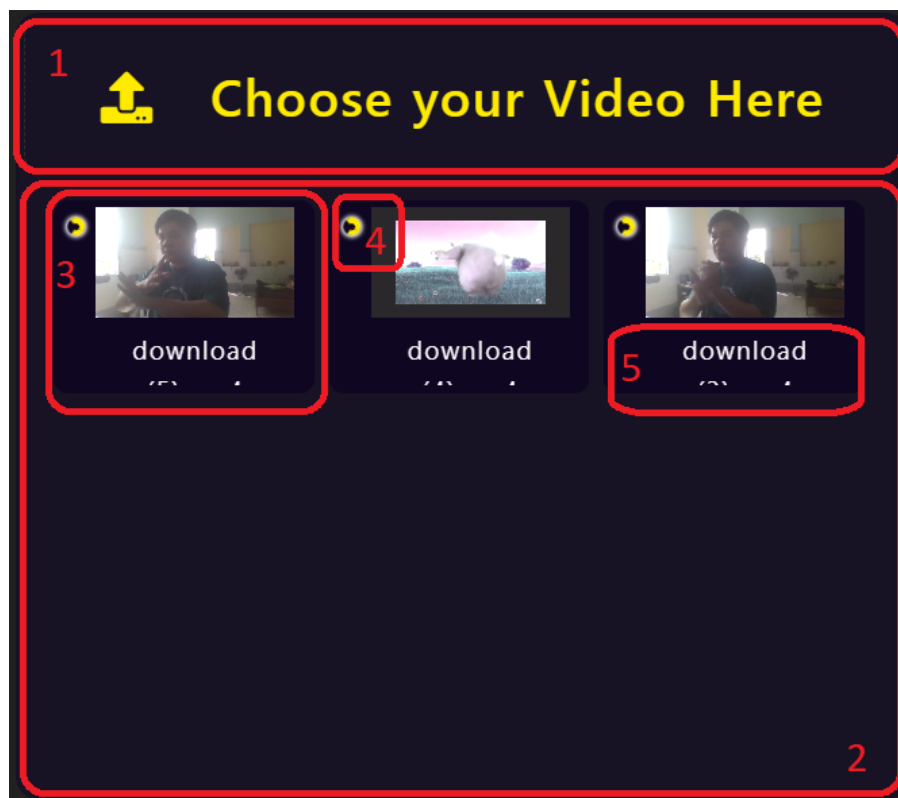
Bảng 21 Bảng mô tả giao diện chung

STT	Tên giao diện	Mô tả
UI01	Đăng tải nội dung video	Giao diện sinh ra với mục đích tạo nên vị trí để người dùng đưa nội dung của họ lên chỉnh sửa

UI02	Video	Giao diện để người dùng có thể xem lại các video, cùng với đó là khu vực tạo nên các điểm cắt.
UI03	Danh sách các điểm cắt video	Phần giao diện thể hiện các điểm cắt video mà người dùng đã tạo nên
UI04	Ghép nội dung video	Giao diện thể hiện các nội dung và vị trí ghép lại của các nội dung được đánh dấu
UI05	Modal setting	Giao diện điều chỉnh lại một số thông tin liên quan tới video cuối cùng

4.3.2.2. Mô tả chi tiết giao diện

a. Giao diện đăng tải nội dung video



Hình 22 Hình ảnh giao diện chi tiết của giao diện đăng tải nội dung video

Bảng 22 Bảng mô tả chi tiết giao diện đăng tải nội dung video

STT	Tên đối tượng	Loại	Mô tả
1	Input video	Thẻ INPUT	Khu vực để chọn video từ trong máy tính, hoặc có thể kéo thả các video
2	Video wrapper	Thẻ Div	Khu vực chứa các video được upload lên
3	Video item	Thẻ button	Thẻ thể hiện video được upload lên, cùng với đó là thumbnail của video đó.
4	Video loading	Thẻ span	Thẻ hiện video đã được gửi lên server hay chưa, khi thẻ này biến mất thì video đã tải lên hoàn thành
5	Video name	Thẻ div	Là khu vực thể hiện tên của video được tải lên video

b. Giao diện video



Hình 23 Hình ảnh giao diện chi tiết của giao diện video



Hình 24 Hình ảnh giao diện chi tiết của giao diện video

Bảng 23 Bảng mô tả chi tiết giao diện video

STT	Tên đối tượng	Loại	Mô tả
1	Video player	Thẻ video	Đây là khu vực phát video sau khi được chọn tại giao diện đăng tải nội dung video, trạng thái ban đầu của video sẽ là tạm dừng
2	Timer	Thẻ div	Đây là vị trí thể hiện video đang được phát với đầu theo số học, định dạng của khu vực này là giờ, phút, giây
3	Time Seeker	Thẻ Input[slider]	Đây là khu vực thể hiện thời gian video đang được phát tới đâu theo phương thức hình học
4	Stop button	Thẻ button	Đây là nút để bấm dừng video lại, video sẽ được tự chuyển về vị trí khung ảnh đầu tiên
5	Play/ Pause button	Thẻ button	Đây là nút làm cho video có thể tạm dừng phát tại vị trí đang phát, và phát tiếp tục tại vị trí đang phát.
6	Previous button	Thẻ button	Đây là nút làm cho video tua ngược lại vị trí 5 giây trước tại vị trí đang được phát
7	Next button	Thẻ button	Đây là nút làm cho video tua tới 5 giây tại vị trí đang được phát
8	Cut function button	Thẻ button	Nút làm chuyển đổi thành time seeker trở thành thanh đánh dấu vị trí.
9	Mute button	Thẻ button	Đây là nút giúp người dùng tắt tiếng nhanh chóng
10	Volume slider	Thẻ input[slider]	Là một thanh slider cho phép người dùng kéo, điều chỉnh âm lượng của video đang phát

11	Start marker	Thẻ input	Là vị trí thể hiện điểm bắt đầu đánh dấu, từ đó có thể xác định vị trí ban đầu của các điểm cắt hoặc lặp lại
12	Double slider	Thẻ input[slider]	Đây là thẻ hỗn hợp, được thiết lập từ hai thanh slider. Mục đích là để người dùng có thể chọn vị trí ban đầu và kết thúc một cách thuận tiện
13	End marker	Thẻ input	Là vị trí thể hiện điểm kết thúc theo định dạng thời gian, mục tiêu là giúp người dùng có thể nhìn thấy chính xác thời điểm được đánh dấu kết thúc
14	Cut button	Thẻ button	Là nút tạo ra các điểm cắt, các điểm cắt, các điểm này sẽ tạo thành từng item riêng biệt và sẽ được gửi sang giao diện danh sách cắt
15	Loop button	Thẻ button	Là nút để tạo ra vị trí lặp đi lặp lại, sau đó khi phát đoạn video thì nó sẽ tự động lặp lại đoạn thời gian đã chọn

c. Giao diện danh sách các điểm cắt

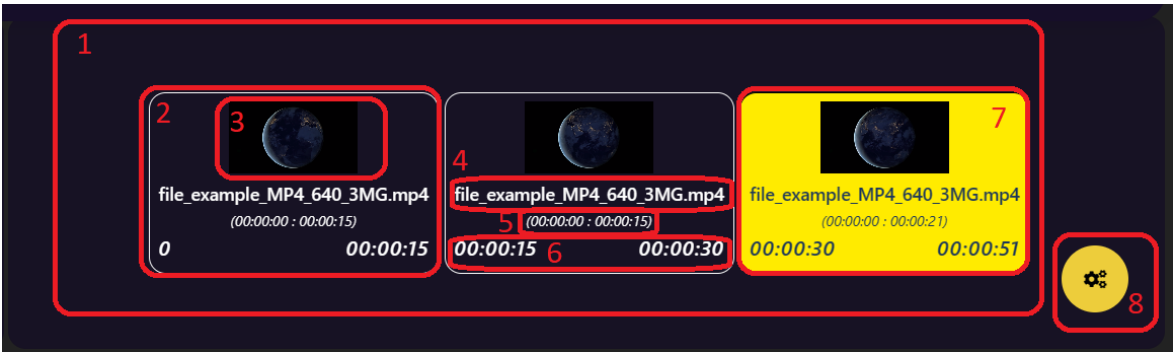


Hình 25 Hình ảnh giao diện chi tiết của giao diện danh sách điểm cắt

Bảng 24 Bảng mô tả chi tiết giao diện danh sách các điểm cắt

STT	Tên đối tượng	Loại	Mô tả
1	Marked cut item	Thẻ div	Là thẻ chứa đựng và thể hiện các thông tin về các điểm cắt
2	Video thumbnail	Video	Là vị trí thể hiện thumbnail của video từ đó có thể dễ dàng xác định video
3	Video name	Div	Vị trí thể hiện tên của video được tạo điểm cắt
4	Cut point	Div	Là vị trí thể hiện thời gian mà điểm cắt tạo ra, bởi điểm đầu (thông số bên phải), và điểm kết thúc (thông số bên trái)
5	Delete button	Thẻ button	Là nút để có thể xóa khỏi danh sách điểm cắt
6	Hover div	Thẻ div	Là hình ảnh của đối tượng 1 khi giữ chuột trên thẻ, tạo thành event hover

d. Giao diện ghép nội dung video



Hình 26 Hình ảnh giao diện chi tiết của giao diện ghép nội dung video

Bảng 25 Bảng mô tả chi tiết giao diện ghép nội dung video

STT	Tên đối tượng	Loại	Mô tả
1	Arrange video wrapper	Thẻ Div	Là khung bao trọn lại toàn bộ các video đã được sắp xếp, khung này sẽ tự canh giữa và mở rộng theo chiều ngang, từ đó sẽ giúp người dùng dễ chú ý hơn.
2	Video item	Thẻ div	Là khung chứa bao lại toàn bộ các thông tin của một video và nó chia cắt các video ra, từ đó ta có thể dễ dàng xác định được vị trí của từng video, ngoài ra ta có thể nhấn vào tag này để xóa đi thông tin trong thẻ wrapper
3	Video thumbnail	Thẻ video	Đây là thẻ video được thiết kế để nó trở thành một ảnh thumbnail cho vị trí video cắt.
4	Video name	Thẻ div	Là vị trí thể hiện tên của đoạn video, đây trở thành một phương thức xác định đúng video cần thay đổi
5	Video point	Thẻ div	Là vị trí thể hiện đoạn video đó được cắt từ đâu tới đâu. Phía dãy số bên trái thể hiện điểm bắt đầu, còn dãy bên phải là hiện kết thúc


6	Video duration	Thẻ div	Là vị trí thể hiện thời gian đoạn video xuất hiện trong video được render, nó thể hiện thời gian đoạn cắt đó được bắt đầu và thời gian đoạn cắt đó kết thúc
7	Video item hover	Thẻ div hover	Thẻ hiện event hover của thẻ video item khi người dùng đưa chuột lại gần tạo ra highlight để người dùng phân biệt thẻ đang được chọn
8	Render option button	Thẻ button	Là nút để mở modal pop up để điều chỉnh các thông số của video trước khi gửi thông tin yêu cầu render, các thông tin như khung hình thể hiện, tốc độ phát.

e. Giao diện modal setting

Request Option 1 X

2 Project Name

3 loadreview

4 

5 Default

6 Default

7 Default

8 Emit

9 Load Preview Video

0 1 2 10

11 Extension Name MP4

12 Video Ratio Landscape

13 Resolution 720P

14 SpeedUp: 2x ☐

15 LutActive: Deactive

16 Send Request

Hình 27 Hình ảnh giao diện chi tiết của giao diện modal setting

Bảng 26 Bảng mô tả chi tiết giao diện modal setting

STT	Tên đối tượng	Loại	Mô tả
1	Close popup button	Button	Là nút để tắt đi bảng pop-up đang hiển thị
2	Project name	Input	Là khung thể người dùng đặt các tên gọi nhớ cho đoạn video được chỉnh sửa
3	Loadreview	Button	là nút để mở hộp thoại xem trước
4	Video Review	Video	Là thẻ video để người dùng xem lại video gồm các điểm cắt ghép thô
5	Hue Rotate slider	Input[range]	Là thanh để thay đổi hue của video chỉ số được thể hiện từ -180 tới 180

6	Saturation slider	Input[range]	Là thanh để thay đổi độ tươi của màu sắc, nó giúp màu trở nên đậm hơn hoặc nhạt hơn
7	Brightness slider	Input[range]	Là thanh để thay đổi độ sáng của màu sắc, nó giúp hình ảnh trên video sẽ sáng hoặc tối hơn
8	Emit button	Button	Là nút để kết xuất màu vào video để đưa cho server render
9	Load Preview Video	Button	Là nút để play video được chỉnh sửa thô
10	Skip point	Button	Là vị trí đặt các nút để tua với các vị trí cắt, trước đó một giây
11	Extension Name	Selection	Là thanh lựa chọn để chọn phần đuôi mở rộng của video sau khi xuất
12	Video ratio selection	Selection	Là thẻ selector để chọn ra hình dạng khung hình cho video xuất ra, là video ngang hay video dọc
13	Video Resolution	Selection	Là thẻ select để lựa chọn độ phân giải của video.
14	Speedup checkbox	Input[checkbox]	Là checkbox để server xác định là có thay đổi tốc độ phát của video lên 2 lần hay không
15	LUT Activate	Thẻ div	Là thẻ div thể hiện việc người dùng đã thêm filter vào video chưa, tại đây có 2 trạng thái là active và deactivate thể hiện là có đặt filter lên hay chưa
16	Send request button	Button	Là nút để có thể gửi toàn bộ thông tin về video được yêu cầu thực hiện lên server, trong đó có cả các thông tin về render trong bảng chọn render option

4.4. THIẾT KẾ XỬ LÝ DỮ LIỆU ĐOẠN VIDEO

Về chương trình chỉnh sửa video này, phần xử lý video được sử dụng nền tảng FFmpeg để tạo nên, nền tảng này cho phép thực hiện các thao tác lên video ở mức độ bitrate từ đó việc tác động lên video sẽ trở nên đơn giản. Tuy nhiên nền tảng này không có một giao diện lý thuyết, để có thể thao tác được ta thường phải thực hiện có đoạn yêu cầu lên nền tảng, các yêu cầu này được viết như các đoạn code, từ đó có thể tương tác lên video.

4.4.1. Thiết kế cách thức cắt video

Bảng 27 Bảng mô tả chi tiết thiết kế cách thức cắt video

Đoạn code	ffmpeg -y -ss startPoint -to endPoint -i fileInputName -c copy fileOutputName	
STT	Thuộc tính	Ý nghĩa
1	ffmpeg	Đây là thuộc tính để gọi nền tảng này ra để thực hiện các thao tác sau đó
2	-y	Có ý nghĩ là đồng ý với tất cả yêu cầu, câu lệnh này được sinh ra để các thông tin có thể ghi đè lên nếu xảy ra sự cố
3	-ss	Là thuộc tính xác định vị trí bắt đầu của điểm cần cắt ở đây sẽ là startPoint
4	-to	Là thuộc tính xác định điểm cuối cùng của đoạn cắt ở đây sẽ là endPoint
5	-i	Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là fileInputName
6	-c	Là thuộc tính chỉ về codec, nó nói về phương thức encodes và decodes trên từng bit data
7	Copy	Là thuộc tính chỉ việc copy ra, và ở đây là copy ra file fileOutputName

4.4.2. Thiết kế cách thức ghép video

Bảng 28 Bảng mô tả chi tiết thiết kế cách thức ghép video

Đoạn code	ffmpeg -y -i fileNameNo1 -i fileNameNo2 -filter_complex "[0]scale= frameRatio :force_original_aspect_ratio=decrease, pad= frameRatio :(ow-iw)/2:(oh-ih)/2, setsar=1[v0];[1]scale= frameRatio :force_original_aspect_ratio=decrease,pad= frameRatio :(ow-iw)/2:(oh-ih)/2, setsar=1[v1];[v0][0:a:0][v1][1:a:0]concat=n=2:v=1:a=1[v][a]" -map "[v]" -map "[a]" fileOutputName	
STT	Thuộc tính	Ý nghĩa
1	-i	Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là fileNameNo1 và fileNameNo2
2	-filter_complex	Là thuộc tính để tạo ra các câu lệnh phức hợp tác động lên video, ta có thể tổ hợp từ 2 câu lệnh lại với nhau, nhưng sau đó cần phải ghép các thông tin lại với nhau
3	[variable]	Đây là thuộc tính nói về vị trí của file được đưa vào thông qua thuộc tính -I, nếu variable là số thì chỉ vị trí của video trong mảng, nếu là chữ thì chỉ thuộc tính tác động lên
4	scale=	Đẩy kích thước của video lên đúng với kích thước của frameRatio
5	force_original_aspect_ratio=decrease	Là lệnh yêu cầu không thay đổi chiều kéo dãn
6	pad= frameRatio :(ow-iw)/2:(oh-ih)/2	Là lệnh yêu cầu tạo ra khung ảnh chính để có thể đặt các video lên, ow là output-width, iw là input-width, oh là output-height và ih là input-height
7	setsar=1[v1];[v0][0:a:0][v1][1:a:0] concat=n=2:v=1:a=1[v][a]	Là đoạn lệnh yêu cầu ghép vị trí tương đối giữa hai khung ảnh, và ghép hai âm thanh lại

8	-map "[v]" -map "[a]"	Nối hai phần video và audio lại và cuối cùng là ghi vào một file tên fileOutputName
---	-----------------------	--

4.4.3. Thiết kế phương thức tăng tốc video

Bảng 29 Bảng mô tả chi tiết thiết kế cách thức tăng tốc video

Đoạn code		ffmpeg -i fileInputName -filter_complex "[0:v]setpts=0.5*PTS[v];[0:a]atempo=2.0[a]" -map "[v]" -map "[a]" fileOutputName
STT	Thuộc tính	Ý nghĩa
1	-i	Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là fileInputName
2	-filter_complex	Là thuộc tính để tạo ra các câu lệnh tác động lên video một cách phức hợp
3	[0:v]setpts=0.5*PTS[v]; [0:a]atempo=2.0[a]	Là đoạn lệnh ghép trong đó setpts làm thay đổi tốc độ video, còn atempo là nén phần âm thanh lại, PTS là tốc độ phát.
4	-map "[v]" -map "[a]"	Nối hai phần video và audio lại và cuối cùng là ghi vào một file tên fileOutputName

4.4.4. Thiết kế phương thức thay đổi màu sắc video

Bảng 30 Bảng mô tả chi tiết thiết kế cách thức thay đổi màu sắc video

Đoạn code		ffmpeg -y -i fileInputName -vf hue=h= filterData.hue :s= filterData.saturation :b= filterData.brightness fileOutputName
STT	Thuộc tính	Ý nghĩa
1	-i	Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là fileInputName
2	-vf	Là thuộc tính để tạo ra các câu lệnh phức tạp nhưng được phân cách bởi “;”, thường được dùng cho nhóm lệnh đơn giản
3	Hue	Là thuộc tính chỉ về màu sắc để tác động lên
4	h	Là thuộc tính chỉ hue rotate được đo bằng độ
5	s	Là thuộc tính chỉ saturation, điểm cân bằng là 1 từ đó sẽ thay đổi tăng giảm để độ tươi của màu thay đổi theo
6	b	Là thuộc tính chỉ brightness, điểm cân bằng là 1 từ đó tăng hoặc giảm sẽ làm cho độ sáng của video thay đổi theo

CHƯƠNG 5. CÀI ĐẶT VÀ KIỂM THỬ

5.1. CÀI ĐẶT

5.1.1. Cài đặt môi trường

Cài đặt NodeJS phiên bản 16.17.1, tại trang chủ chính thức của NodeJS: <https://nodejs.org/en/download/releases/>

Tải xuống FFmpeg tại trang chủ: <https://ffmpeg.org/download.html>

Tải xuống chương trình tại: <https://github.com/thedeadsheep/ang-VideoEditing>

5.1.2. Cài đặt phần mềm

Front-end:

- Mở thư mục của chương trình vừa tải từ trang github
- Mở command line tại chính thư mục đó và nhập lệnh:
 - (1) Chạy **npm install (hoặc npm i)** để cài đặt các thư viện cần thiết để chạy chương trình
 - (2) Chạy **npm start** để khởi động chương trình, về mặc định chương trình sẽ chạy tại **localhost:4200/**

Back-end:

- Tại thư mục project, ta truy cập vào thư mục BE.
- Tại thư mục BE ta tạo mới một thư mục **uploads**, đây là thư mục sẽ chứa các đoạn video được người dùng đăng tải lên server.
- Cũng tại thư mục BE này, ta mở command line là thực hiện các đoạn lệnh sau:
 - (1) Chạy **npm install (hoặc npm i)** để cài đặt các thư viện cần thiết để chạy chương trình
 - (2) Chạy **npm start** để khởi động chương trình, về mặc định chương trình sẽ chạy tại localhost:3000.

FFmpeg:

Sau khi download được phiên bản bất kỳ trên trang chủ của FFmpeg. Đối với Windows ta thực hiện **giải nén file tải về** đặt thư mục này tại một vị trí bất kỳ, truy cập vào **Edit the system Environment variables, Environment Variables**, tại vị trí **Path** ta thêm vị trí thư mục giải nén của ffmpeg, sau đó mở commandline nhập **ffmpeg** để kiểm tra ffmpeg đã có thể hoạt động chưa.

Truy cập vào chương trình:

- Ta mở một web browser bất kỳ và truy cập vào địa chỉ sau: <http://localhost:4200/> để truy cập vào chương trình

5.2. KIỂM THỬ

5.2.1. Kiểm thử chức năng login

Bảng 31 Bảng kiểm thử chức năng login

Testcase	Test title	Steps	Data	Expected result	Actual result	Pass
TClogin1	Đăng nhập thành công	(1) Chuyển vào trang đăng nhập (2) Nhập email (3) Bấm nút getPassword (4) Nhận password từ email (5) Nhập password (6) Bấm nút đăng nhập	Tranviet-mar15-@gmail.com	Đăng nhập thành công	Đăng nhập thành công, chuyển qua trang dashboard	pass

5.2.2. Kiểm thử chức năng sign up

Bảng 32 Bảng kiểm thử chức năng sign up

Testcase	Test title	Steps	Data	Expected result	Actual result	Pass
TCsignin1	Đăng ký thành công	(1) Chuyển vào trang đăng ký (2) Nhập email: (3) Nhập nickname “Shirou” (4) Bấm nút SignUp (5) Kiểm tra email (6) Click vào link xác thực email	Tranviet-mar15-@gmail.com	Email có thư xác nhận	Có email xác nhận	Pass

5.2.3. Kiểm thử chức năng upload

Bảng 33 Bảng mô tả kiểm thử chức năng upload video

Testcase	Test title	Steps	Data	Expected result	Actual result	Pass
TCupload1	Đăng tải video bất kỳ	(1) Mở hộp thoại đăng tải (2) Chọn video 1.mp4 (3) Bấm nút đăng tải.	Video 1.mp4	Video được upload lên, và không xảy ra sự cố	Video được upload lên, và không xảy ra sự cố	Pass
TCupload2	Đăng tải video, lại một video đã được upload	(1) Mở hộp thoại đăng tải (2) Chọn video 1.mp4 (3) Bấm nút đăng tải.	Video 1.mp4	Video mới được chọn không hiển thị lên	Video sau không xuất hiện	Pass

		(4) Lập lại thao tác mở hộp thoại đăng tải. (5) Chọn lại video 1.mp4 (6) Bấm nút đăng tải		chương trình		
TCupload3	Kéo thả các video chương trình	(1) Mở thư mục chứa video cần đăng tải. (2) Nhấn kéo thả các video vào chương trình	Video 1.mp4	Các video được thả vào chương trình, xuất hiện trên giao diện	Video xuất hiện trên chương trình	Pass

5.2.4. Kiểm thử chức năng send request và render video.**Bảng 34 Bảng kiểm thử chức năng send request và render video.**

Testcase	Test title	Steps	Data	Expected result	Actual result	Pass
TCrender01	Yêu cầu render hai video tương đồng	<p>Chọn video từ máy tính</p> <p>Click chuột vào video được đăng tải</p> <p>Chọn sang chế độ động đánh dấu</p> <p>Đánh dấu video từ 0 giây tới 4 giây.</p> <p>Đánh dấu video từ 6 giây tới 12 giây</p> <p>Sắp xếp 2 đoạn cắt lại</p> <p>Gửi yêu cầu render video</p>	Video 1.mp4	Video trả về dài 10 giây có khung hình tương đồng với video gốc	Video trả về với độ dài 10 giây	Pass
TCrender02	Yêu cầu render một video ngang thành một video dọc	<p>Chọn video từ máy tính</p> <p>Click chuột vào video được đăng tải</p> <p>Chọn sang chế độ động đánh dấu</p> <p>Đánh dấu video từ 0 giây tới 4 giây.</p> <p>Chọn đoạn cắt.</p> <p>Gửi yêu cầu xử lý video với video ratio là landscape.</p>	Video 2.mp4	Video trả về có độ dài là 4 giây và khung hình video sẽ bị bóp lại thành chiều dọc	Video dọc có độ dài 4 giây	Pass

CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1. TỔNG KẾT

Sau khoảng 15 tuần, từ lúc nhận đề tài cho tới khi bắt tay đặt nền móng đầu tiên cho chương trình, bản thân em đánh giá chương trình đã đạt được những mục tiêu ban đầu đề ra, đó là có khả năng cắt ghép các đoạn video lại với nhau. Cùng đó qua việc tìm hiểu về đề tài, bản thân em tiếp thu được rất nhiều kiến thức từ trong việc lập trình và kinh nghiệm trong việc chỉnh sửa video. Ngoài ra từ đó bản thân em có được thêm nhiều kiến thức trong việc chuyển đổi một phần mềm từ nền tảng này lên một nền tảng khác, từ đó trau dồi kỹ năng phân tích và giải quyết vấn đề, hiểu được giới hạn của một số nền tảng.

6.2. KẾT LUẬN

6.2.1. Về mặt lý thuyết

Sau thời gian thực hiện đề tài, bản thân em đã tìm hiểu được một vài điều sau đây:

- Sử dụng Angular và NodeJS để tạo nên một phần mềm ứng dụng chạy trên nền tảng web
- Tận dụng giới hạn của webapp để tối đa hóa hiệu năng
- Cách tạo ra một server điều khiển một máy tính cá nhân.
- Ứng dụng phần học của môn tương tác máy tính vào chương trình để thay đổi UX cho hợp với một application chạy trên web.

6.2.2. Về mặt ứng dụng

WebApplicaton chỉnh sửa video đã được hoàn thành một cách tương đối, có thể sử dụng nó như một công cụ để cắt các đoạn video một cách nhanh chóng, ngoài ra có thể ghép các đoạn lại thành một video hoàn chỉnh theo ý thích

6.2.3. Ưu điểm

Chương trình đã thay đổi một ít về mặt UX so với các phần mềm khác, để trở nên thân thiện với người dùng web hơn. Ngoài ra, về việc giảm thiểu việc tương tác với server một cách tối đa đã làm chương trình trở nên nhanh hơn tương đối nếu để so với một webapp, tận dụng được giới hạn của HTML, làm nó trở nên gọn gọn hơn và dùng ít tài nguyên của máy tính hơn.

6.2.4. Nhược điểm

Về nhược điểm, chương trình thực chất có thời gian phát triển rất ít, vì phần lớn thời gian là để tìm phương pháp và tài liệu liên quan, cùng với đó toàn bộ được một mình

em phát triển nên nó trở nên chậm chạp. Ngoài ra phần mềm hiện tại rất đơn sơ, tuy có thể cắt ghép được nhưng một phần cũng không thể nói hoàn chỉnh vì các testcase được người phát triển trực tiếp nghĩ ra nên không thể bao quát được toàn bộ.

6.3. HƯỚNG PHÁT TRIỂN

- Thêm vào một số chức năng tùy chỉnh thông số về mặt kỹ thuật như bitrate.
- Thêm vào phần thêm âm nhạc đè lên phần âm thanh, hoặc tách riêng phần âm thanh.
- Cải thiện giao diện trở nên thân thiện hơn.
- Cải thiện hiệu xuất ghép video.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Web cơ bản, Thẻ <video> trong HTML, [Online]. Available: <https://webcoban.vn/html/the-video-trong-html-nhung-mot-trinh-xem-phim-vao-trang-web.html>

[2] @quandv Viblo, Sử dụng @Input() và @Output() trong angular, [Online]. Available: <https://viblo.asia/p/su-dung-atinput-va-atoutput-trong-angular-3P0IPJP8Kox>

Tiếng Anh

[3] FFmpeg, Document, [Online]. Available: <https://ffmpeg.org/ffmpeg.html>

[4] gopinav, Angular-Tutorial, [Online] Available: <https://github.com/gopinav/Angular-Tutorials>

[5] Samer Buna, Node.js Child Processes: Everything you need to know, [Online]. Available: <https://www.freecodecamp.org/news/node-js-child-processes-everything-you-need-to-know-e69498fe970a/>